Ausbildungsunterlage für die durchgängige Automatisierungslösung Totally Integrated Automation (T I A)

MODUL E11

Radio Frequency Identification (RFID) mit SIMATIC S7-300F-2 PN/DP und RF180C

Diese Unterlage wurde von der Siemens AG, für das Projekt Siemens Automation Cooperates with Education (SCE) zu Ausbildungszwecken erstellt.

Die Siemens AG übernimmt bezüglich des Inhalts keine Gewähr.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts ist innerhalb öffentlicher Aus- und Weiterbildungsstätten gestattet. Ausnahmen bedürfen der schriftlichen Genehmigung durch die Siemens AG (Herr Michael Knust michael.knust@siemens.com). Zuwiderhandlungen verpflichten zu Schadensersatz. Alle Rechte auch der Übersetzung sind vorbehalten, insbesondere für den Fall der Patentierung oder GM-Eintragung.

Wir danken der Fa. Michael Dziallas Engineering und den Lehrkräften von beruflichen Schulen sowie weiteren Personen für die Unterstützung bei der Erstellung der Unterlage

SEITE:

1	Vorwort	4
2	Hinweise zum Einsatz der CPU 315F-2 PN/DP	6
3	Hinweise zu den SIMATIC RFID-Komponenten	7
4	RFID Grundlagen	8
5	Inbetriebnahme eines RFID-Projektes mit CPU 315F-2 PN/DP und RF180C	9
5.1	Neues Projekt anlegen	10
5.2	Hardware Konfigurieren	12
5.3	Gerätenamen zuweisen	19
5.4	UDT Bausteine und FB45 einfügen	21
5.5	Datenbausteine erstellen	22
5.6	Programmierung von Neustart und Wiederanlauf	31
5.7	FC11 Funktion für einen Befehl bzw. Auftrag	
5.8	Grundlagen zu den Eingaben am Kommando-Baustein FC11	34
5.9	Befehlskette	
5.10	Grundlagen zum FB45 und DB45	
5.11	FB10 SLG_Steuerungsprogramm	47
5.12	FB1 Steuerungsprogramm	50
5.13	OB1 Programmaufruf	55
5.14	Variablentabelle STATUS_SLG_1	
5.15	Variablentabelle STATUS_SLG_2	57
5.16	Symboltabelle	
5.17	Bausteinordner	
5.18	Datensicht des DB49	

Die folgenden Symbole führen durch dieses Modul:



1 VORWORT

1

Das Modul E11 ist inhaltlich der Lehreinheit ,IT-Kommunikation mit SIMATIC S7' zugeordnet.



Lernziel:

Der Leser soll in diesem Modul lernen wie die Vernetzung und der Datenaustausch zwischen SPSen und RFID-Komponenten aufgebaut wird.

Als SPS-Steuerung wird eine CPU 315F-2 PN/DP und als Radio Frequency Identification (RFID) ein SIMATIC RFID System eingesetzt. Die RFID-Komponenten bestehen aus einem Anschaltmodul RF180C (ASM) mit Schreib-Lese-Gerät RF310R (Reader oder SLG) und verschiedene Mobile-Datensysteme wie RF340T, RF350T, RF360T oder ISO Moby D MDS D124 (Transponder oder MDS). Die Vernetzung zwischen SPS und SIMATIC RF180C erfolgt über PROFINET. Das Modul zeigt die prinzipielle Vorgehensweise zur Inbetriebnahme anhand eines kurzen Beispiels.

Voraussetzungen:

Für die erfolgreiche Bearbeitung dieses Moduls wird folgendes Wissen vorausgesetzt:

- Kenntnisse in der Handhabung von Windows
- Grundlagen der SPS- Programmierung mit STEP 7 (z.B. Modul A3 ,Startup' SPS- Programmierung mit STEP 7)
- Grundlagen der Netzwerktechnik (z.B. Anhang V Grundlagen der Netzwerktechnik)

1

Benötigte Hardware und Software

- 1 PC, Betriebssystem Windows XP Professional mit SP2 oder SP3 / Vista 32 Bit Ultimate und Business / Server 2003 SP2 mit 600MHz (nur XP) / 1 GHz und 512MB (nur XP) / 1 GB RAM, freier Plattenspeicher ca. 650 - 900 MB, MS-Internet-Explorer 6.0 und Netzwerkkarte
- 2 Software STEP 7 V 5.4
- SPS SIMATIC S7-300 mit CPU 315F-2 PN/DP und mindestens einer digitalen Ein- und Ausgabebaugruppe.
 Beispielkonfiguration:
 - Netzteil: PS 307 2A
 - CPU: CPU 315F-2 PN/DP
 - Digitale Eingänge: DI 16x DC24V
 - Digitale Ausgänge: DO 16x DC24V / 0,5 A
- SIMATIC RF180C ASM (Anschaltmodul) RF310R oder RF340R SLG (Schreib-Lese-Gerät) verschiedene Transponder MDS (Mobiles-Daten-System) vom Typ: RF340T (8 KB) RF350T (32 KB) RF360T (64 KB) ISO Moby D MDS D124 (112 Byte)
- 5 Ethernet- Verbindung zwischen PC, CPU 315F-2 PN/DP und RF180C



mit Reader RF310R (SLG) und Transponder RF340T (MDS)

2 HINWEISE ZUM EINSATZ DER CPU 315F-2 PN/DP



Die CPU 315F-2 PN/DP ist eine CPU die mit 2 integrierten Schnittstellen ausgeliefert wird.

- Die erste Schnittstelle ist eine kombinierte MPI/PROFIBUS-DP– Schnittstelle, die am PROFIBUS DP als Master oder Slave für den Anschluss von dezentraler Peripherie/Feldgeräten mit sehr schnellen Reaktionszeiten eingesetzt werden kann. Des Weiteren kann Die CPU hier über MPI oder auch über PROFIBUS DP programmiert werden
- Die zweite Schnittstelle ist eine integrierten PROFINET- Schnittstelle.
 Diese ermöglicht den Einsatz der CPU als PROFINET IO- Controller für den Betrieb von dezentraler Peripherie an PROFINET. Über diese Schnittstelle kann die CPU ebenfalls programmiert werden!
- An beiden Schnittstellen können auch fehlersichere Peripheriegeräte eingesetzt werden.



Hinweise:

- In diesem Modul wird die CPU 315F-2 PN/DP als Steuerung für den Datenaustausch eines SIMATIC RFID-Systems am PROFINET eingesetzt.
- Zum Betrieb dieser CPU ist eine Micro Memory Card erforderlich!
- Die Adressen der Ein- und Ausgangsbaugruppen können bei dieser CPU parametriert werden.

1

3 HINWEISE ZU DEN SIMATIC RFID-KOMPONENTEN

Das Kommunikationsmodul RF180C ist eine Baugruppe für den Betrieb von RFID-Komponenten über PROFINET IO an beliebigen Steuerungen.

Am RF180C können bis zu 2 Reader (SLG) parallel betrieben werden. Der Anwender kann parallel an 2 Readern (SLG) einen Befehl starten (FB 45 bei Betrieb an einer SIMATIC S7). Der Zugriff auf die Tag-Daten erfolgt über die physikalische Adressierung des Tags. In der SIMATIC S7 steht hierfür der FB 45 zur Verfügung. Der FB 45 stellt dem S7-Anwender eine einfach handhabbare Schnittstelle mit mächtigen Befehlen zur Verfügung (Bearbeitung eines kompletten Tags mit einem Befehl; Befehlskettung; S7-Datenstrukturen über UDTs).



Die Einbindung des RF180C in die Hardwarekonfiguration erfolgt über eine GSDML-Datei. Danach kann das RF180C über HW-Konfig des SIMATIC-Managers konfiguriert werden. Die GSDML-Datei befindet sich auf der CD "RFID Systems".



Weitere Informationen finden Sie in der Betriebsanleitung "RFID-Systeme Kommunikationsmodul RF180C".

4 RFID GRUNDLAGEN



Der englische Begriff "**Radio Frequency Identification**" (**RFID**) bedeutet im Deutschen "**Identifizierung mit Hilfe von elektromagnetischen Wellen**". RFID ermöglicht die automatische Identifizierung und Lokalisierung von Gegenständen und Lebewesen und erleichtert damit erheblich die Erfassung und Speicherung von Daten. Ein RFID-System besteht aus einem Transponder, der sich am oder im Gegenstand bzw. Lebewesen befindet und diese kennzeichnet, sowie einem Lesegerät zum Auslesen der Transponder-Kennung. Das Lesegerät enthält eine Software (ein Mikroprogramm), das den eigentlichen Leseprozess steuert, und eine RFID-Middleware mit Schnittstellen zu weiteren EDV-Systemen und Datenbanken.

In der Regel erzeugt das Lesegerät ein elektromagnetisches Hochfrequenzfeld geringer Reichweite, vorzugsweise mit Induktionsspulen. Damit werden nicht nur Daten übertragen, sondern auch der Transponder mit Energie versorgt. Nur wenn größere Reichweiten erzielt werden sollen, werden aktive Transponder mit eigener Stromversorgung eingesetzt. Meist wird die Frequenz 13,56 MHz benutzt (RF300, ISO). Das Lesegerät (Reader) erzeugt ein hochfrequentes elektromagnetisches Wechselfeld, welches die Antenne des RFID-Transponders (RFID-Tag) beleuchtet. In der Antennenspule entsteht, sobald sie in das elektromagnetische Feld kommt, ein Induktionsstrom. Dieser Strom wird gleichgerichtet und damit ein Kondensator als Kurzzeitspeicher aufgeladen, welcher für den Lesevorgang die Stromversorgung des Chips besorgt. Diese Versorgung übernimmt bei aktiven Tags eine eingebaute Batterie. Der so aktivierte Mikrochip im RFID-Tag decodiert die vom Lesegerät gesendeten Befehle. Die Antwort codiert und moduliert dieser "Reader" in das eingestrahlte elektromagnetische Feld durch Feldschwächung im kontaktfreien Kurzschluss oder gegenphasige Reflexion des vom Lesegerät ausgesendeten Feldes. Damit sendet das Tag seine eigene unveränderliche Seriennummer, weitere Nummern des gekennzeichneten Objekts oder andere vom Lesegerät abgefragte Daten. Der Transponder sendet selbst kein Feld aus, sondern verändert nur das elektromagnetische Sendefeld des Readers.



Schalter des Transponders offen, Antenne steht auf "Absorption" (Rückgabewert 0).



Schalter des Transponders geschlossen, Antenne steht auf "Reflektieren" (Rückgabewert 1). (Quelle: Wikipedia)



INBETRIEBNAHME EINES RFID-PROJEKTES MIT CPU 315F-2 PN/DP UND RF180C

Im Folgenden wird die Inbetriebnahme eines RFID-Projektes beschrieben. Dabei wird als SIMATIC S7-300 Station eine CPU 315F-2 PN/DP eingesetzt. Im Steuerungsprogramm der CPU muss eine Datenstrukur über einen Funktionsbaustein-Aufruf (FB45) mit Datenbausteinen und eingebetteten UDT Datentypen erstellt werden. In unserem Beispiel sollen pro Reader folgende Aktionen durchgeführt werden können:

- Schreiben von Daten auf den Transponder (DB 48 der CPU zu MDS).
- Lesen der Daten von dem Transponder (MDS zu DB48 der CPU).
- Lesen der Transponder-Status Informationen (MDS zu DB49 der CPU).
- Lesen der Reader-Status Informationen (SLG zu DB50 der CPU).

Die Datenstruktur mit den deutschsprachigen UDTs hat folgenden Aufbau:



Zum Erstellen der Datenstruktur müssen die notwendigen UDT Datentypen in das Step 7 Projekt und der Funktionsbaustein FB45 importiert werden. Ein Beispielprogramm mit den Bausteinen befindet sich auf der CD "RFID Systems".



Weitere Informationen zu der Datenstruktur finden Sie im Funktionshandbuch "RFID Systems FB45"

5.1 Neues Projekt anlegen



 Das zentrale Werkzeug in STEP 7 ist der ,SIMATIC Manager', der hier mit einem Doppelklick aufgerufen wird. (→ SIMATIC Manager)



 STEP 7- Programme werden in Projekten verwaltet . Ein solches Projekt wird nun angelegt (→ Datei → Neu)

icel zielsystelli Alisichic Extras Penster Hille	
Neu	Ctrl+N
Assistent 'Neues Projekt'	
Offnen	Ctrl+O
S7-Memory Card	•
Memory Card-Datei	•
Löschen	
Reorganisieren	
Verwalten	
Archivieren	
Dearchivieren	
Seite einrichten	
1 PROFIsafe02 (Projekt) F:\0_S7_Projekte\PROFIs_2	
2 Profi Umrichter (Projekt) F:\0_57_Projekte\Profi_Um	
3 Erreichbare Teilnehmer PROFIBUS	
Beenden	Alt+F4

3. Dem Projekt wird nun der **,Name' ,RFID_RF180C'** gegeben (\rightarrow RFID_RF180C \rightarrow OK)

	Ablagepfad 🔨				
Abscherv_Vorlage	C:\SIEMENS\S7Proj\Absche_4				
Abschervorrichtung	C:\Program Files\Siemens\Step7\S				
Abschervorrichtung2	C:\SIEMENS\S7Proj\Absche_2				
🞒 anlagensim_neu	C:\Program Files\Siemens\Step7\S				
🞒 A-Projekt	C:\Program Files\Siemens\Step7\S				
AS_I_Busplatz	C:\SIEMENS\S7Proj\AS_I_Buspla				
RAC I Duzelste MDI	CACIEMENICAC7DiaNAC I DuaME				
In aktuelles Multiprojekt einfi ame:	iigen Typ:				
RFID_RF180C	Proiekt				
Ablageort (Pfad) :					
blageort (Pfad) :	F-Bibliothek				



 Markieren Sie Ihr Projekt und fügen Sie ein ,Industrial Ethernet- Subnetz' ein (→ RFID_RF180C → Einfügen → Subnetz → Industrial Ethernet).

<u>.</u>	🖉 SIMATIC Manager - RFID_RF180C									
Date	i Bearbeiten	Einfügen Zielsystem	Ansicht Extras Fenster Hilfe							
D	🖻 🔡 🛲	Station	🕨 🖳 🖳 📖 👘 🛛 < Kein Filter > 💿 🏹 🞇 🚳 📆							
É.	1	Subnetz	▶ 1 MPI							
		Programm	2 PROFIBUS							
		S7-Software	3 Industrial Ethernet							
		S7-Baustein								
	REID_RE18	M7-Software	nemens vstep / vs / projvkr ID_R_3							
·	BFID_RF	Symboltabelle	I) Globales Schriftfeld							
		Textbibliothek	•							
		Externe Quelle								
		WinCC flexible RT	•							
		Globale Deklaratione	ien 🕨							

5. Dann wird eine **,SIMATIC 300-Station'** eingefügt. (→ Einfügen → Station → SIMATIC 300-Station)



5.2 Hardware Konfigurieren



6. Konfigurationswerkzeug für die ,**Hardware'** mit einem Doppelklick öffnen. (→ Hardware)



7. Hardwarekatalog durch einen Klick auf das Symbol , D, öffnen. (→ D)
,Profilschiene' mit einem Doppelklick einfügen (→ SIMATIC 300 → RACK-300 → Profilschiene).





Hinweis

Danach wird automatisch eine Konfigurationstabelle für den Aufbau des Racks 0 eingeblendet.



8. Aus dem Hardwarekatalog können nun alle Baugruppen ausgewählt und in der Konfigurationstabelle eingefügt werden, die auch in Ihrem realen Rack gesteckt sind. Dazu müssen Sie auf die Bezeichnung der jeweiligen Baugruppe klicken, die Maustaste gedrückt halten und per Drag & Drop in eine Zeile der Konfigurationstabelle ziehen.

Wir beginnen mit dem Netzteil ,**PS 307 5A'.** (\rightarrow SIMATIC 300 \rightarrow PS-300 \rightarrow PS 307 5A)





Hinweis

Falls Ihre Hardware von der hier gezeigten abweicht, so müssen Sie einfach die entsprechenden Baugruppen aus dem Katalog auswählen und in Ihr Rack einfügen. Die Bestellnummern der einzelnen Baugruppen, die auch auf den Komponenten stehen, werden in der Fußzeile des Katalogs angezeigt.



9. Im nächsten Schritt ziehen wir die ,CPU 315F-2 PN/DP' auf den zweiten Steckplatz.
 Dabei können Bestellnummer und Version der CPU auf der Front der CPU abgelesen werden.
 (→ SIMATIC 300 → CPU-300 → CPU 315F-2 PN/DP → 6ES7 315-2FH13-0AB0 → V2.6)

	HW Konfi	g - [SIMATIC 300(1) (Kon	figuration) RFID_RF180	C]						_	
mnl	Station Be	earbeiten Einfügen Zielsyste	em Ansicht Extras Fenster	Hilfe							×
) 🖻 🚰 (# \$# \$ \$ 6. •	🛍 🋍 📳 🗖 👯 🕅								
	_							^			
	30) UF	} 						=	Suchen:		nt ni
	1	PS 307 5A							5 0		
	3								Profil:	Standard	-
	4									CPU-300	^
	5									🕀 🧰 CPU 312	
	6									E CPU 312 IFM	
	7									E CPU 312	
	8										
	10									🗄 🦲 CPU 313C-2 DP	
	11									🗄 🦲 CPU 313C-2 PtP	=
								~		🗄 🧰 CPU 314	_
<								>		🗄 🧰 CPU 314 IFM	
										🕀 🦲 CPU 314C-2 DP	
	(0)	UR								H 🔄 LPU 314L-2 PtP	
		(maine)	1	1	(E CPU 315-2 DP	
	Steckplatz	Baugruppe	Bestellnummer	Firmware	MPI-Adresse	E	A	K		E CPU 315-2 PN/DP	
	1	PS 307 5A	6ES7 307-1EAUU-UAAU							🗄 🦲 CPU 315F-2 DP	
	2									🗄 🦲 CPU 315F-2 PN/DP	
	4									😟 🧰 6ES7 315-2FH10	HOABC
	5									🖻 🧰 📴 6ES7 315-2FH13	-OABC
	6									V2.3	
113	7									V2.5	_
	8									V2.6	¥
	9										
	10								6ES7 31	5-2FH13-0AB0	<u>^</u> ₹ <u><</u>
- 1	11								PROFIN	ET Anschluss: S7-	
									Kommun	ikation (ladbare FBs/FCs);	~
ı Drüc	ken Sie F1,	um Hilfe zu erhalten.							0		Änd

 Beim Eintragen der CPU erscheint folgendes Fenster, in dem Sie der CPU 315F-2 PN/DP eine ,IP- Adresse' zuordnen, die ,Subnetzmaske' festlegen und das bereits erstellte ,Ethernet'-Netz auswählen müssen. Optional kann für Netzübergreifende Kommunikation auch eine ,Router- Adresse' ausgewählt werden. Bestätigen Sie Ihre Eingaben mit ,OK' (→ IP- Adresse: 192.168.0.100 → Subnetzmaske: 255.255.255.0 → Ethernet(1) → Keinen Router verwenden → OK)

Eigenschaften - Ethernet Schnittstelle PN-IO (R0/S2.2)								
Allgemein Parameter								
	Bei Anwahl eines Subnetzes werden die nächsten freien Adressen vorgeschlagen							
IP-Adresse: 192.168.0.100	Netzübergang							
Subnetzmaske: 255.255.255.0	Keinen Router verwenden							
	Adresse: 192.168.0.100							
Subnetz:								
nicht vernetzt Ethernet(1)	Neu							
	Eigenschaften							
	Löschen							
ОК	Abbrechen Hilfe							



Hinweise zur Vernetzung am Ethernet (Weitere Informationen im Anhang V der Ausbildungsunterlage):

MAC- Adresse:

Die MAC-Adresse besteht aus einem festen und einem variablen Teil. Der feste Teil ("Basis-MAC-Adresse") kennzeichnet den Hersteller (Siemens, 3COM, ...). Der variable Teil der MAC-Adresse unterscheidet die verschiedenen Ethernet-Teilnehmer und sollte weltweit eindeutig vergeben werden. Auf jeder Baugruppe ist eine werksseitig vorgegebene MAC- Adresse aufgedruckt.

Wertebereich für IP-Adresse:

Die IP-Adresse besteht aus 4 Dezimalzahlen aus dem Wertebereich 0 bis 255, die durch einen Punkt voneinander getrennt sind; z.B. 141.80.0.16

Wertebereich für Subnetzmaske:

Diese Maske wird verwendet, um erkennen zu können, ob ein Teilnehmer bzw. dessen IP- Adresse zum lokalen Subnetz gehört oder nur über einen Router erreichbar ist.

Die Subnetzmaske besteht aus 4 Dezimalzahlen aus dem Wertebereich 0 bis 255, die durch einen Punkt voneinander getrennt sind; z.B. 255.255.0.0

Die 4 Dezimalzahlen der Subnetzmaske müssen in ihrer binären Darstellung von links eine Folge von lückenlosen Werten "1" und von rechts eine Folge von lückenlosen Werten "0" enthalten. Die Werte "1" bestimmen den Bereich der IP-Adresse für die Netznummer. Die Werte "0" bestimmen den Bereich der IP-Adresse für die Teilnehmeradresse.

Beispiel:

richtige Werte:	255.255.0.0 Dezimal = 1111 1111.1111 1111.0000 0000.0000 0000 Binär
	255.255.128.0 Dezimal = 1111 1111.1111 1111.1000 0000.0000 0000 Binär
	255.254.0.0 Dezimal = 1111 1111.1111 1110.0000 0000.0000.00
falscher Wert:	255.255.1.0 Dezimal = 1111 1111.1111 1111.0000 000 1 .0000 0000 Binär

Wertebereich für Adresse des Netzübergangs (Router):

Die Adresse besteht aus 4 Dezimalzahlen aus dem Wertebereich 0 bis 255, die durch einen Punkt voneinander getrennt sind; z.B. 141.80.0.1.

Zusammenhang IP-Adressen, Adresse des Routers und Subnetzmaske:

Die IP-Adresse und die Adresse des Netzübergangs dürfen nur an den Stellen unterschiedlich sein, an denen in der Subnetzmaske "0" steht. Beispiel:

Sie haben eingegeben: für Subnetzmaske 255.255.255.0; für IP-Adresse 141.30.0.5 und für die Adresse des Routers 141.30.128.1.

Die IP-Adresse und die Adresse des Netzübergangs dürfen nur in der 4. Dezimalzahl einen unterschiedlichen Wert haben. Im Beispiel ist aber die 3. Stelle schon unterschiedlich. Im Beispiel müssen Sie also alternativ ändern:

- die Subnetzmaske auf: 255.255.0.0 oder
- die IP- Adresse auf: 141.30.128.5 oder
- die Adresse des Netzübergangs auf: 141.30.0.1



 Im nächsten Schritt ziehen wir das Eingangsmodul für 16 Eingänge auf den vierten Steckplatz. Dabei kann die Bestellnummer des Moduls auf der Front abgelesen werden. (→ SIMATIC 300 → DI-300 → SM 321 DI16xDC24V).





Hinweis

Steckplatz Nr. 3 ist für Anschaltungsbaugruppen reserviert und bleibt daher leer. Die Bestellnummer der Baugruppe, wird in der Fußzeile des Katalogs angezeigt.



 Im nächsten Schritt ziehen wir das Ausgangsmodul für 16 Ausgänge auf den fünften Steckplatz. Dabei kann die Bestellnummer des Moduls auf der Front abgelesen werden. (→ SIMATIC-300 → DO-300 → SM 322 DO16xDC24V/0,5A).





Hinweis

Die Bestellnummer der Baugruppe, wird in der Fußzeile des Katalogs angezeigt.

 Nun muss noch der PROFINET- Gerätename auf PN-IOx100 geändert werden. Wählen Sie ,**PN-IO**' mit einem Doppelklick. (→ PN-IO,→ PN-IOx100,→ OK)

Eigenschaften - PN-IU (RU/SZ.Z)									
Allgemein Adressen PROFINET Synchronisation Uhrzeitsynchronisation									
Kurzbezeichnung:	PN-IO								
Gerätename: PN-I0x100									
Schnittstelle			1						
Тур:	Ethernet								
Gerätenummer:	0								
Adresse:	192.168.0.100								
Vernetzt:	ja	Eigenschaften							
Kommentar:			1						
					~				
ОК				Abbrechen	Hilfe				



14. Ziehen Sie den PROFINET-IO-System (100) Strang nach rechts und fügen Sie aus dem Ordner PROFINET IO die SIMATIC RFID Baugruppe RF180C durch Ziehen auf den Strang ein Sollte die Baugruppe RF180C noch nicht zur Auswahl stehen muss sie erst über Menü "Extras" GSD-Dateien instalieren von einem Datenträger eingefügt werden.

🖳 HW Konfig - [SIMATIC 300(1) (Konfi	guration) RFID_RF180C]					
🛄 Station Bearbeiten Einfügen Zielsystem	Ansicht Extras Fenster Hilfe					_ @ ×
D 🚅 🔓 🖩 📲 🖓 🖨 🖻 💼 🎪	🛍 👔 🗖 🐮 🕅					
CO) UR CPU 315F-2 P X7 MPI/DP X2 P7 Pol 7 A DI16xDC24V 5 D016xDC24V/0 6 7 8 W	Ethernet(1): PROFINE	1-10-System (100 (1) RF180C (1) RF180C (1) RF180C (1) RF180C (1) RF180C	<u>"</u>		× • • • • • • • • • • • • • • • • • • •	Sucher: At Ai Profit Standard PROFIBUS-DP PROFIBUS-PA PROFIDUS-PA PROFINET IO Gateway HMI Ident Systems SIMATIC FFID SIMATIC FFID SIMATIC FFID
(1) RF180Cx110 Steckplatz Baugruppe <i>O RF180Cx110</i> <i>1 Anschluss zweier Reader</i>	Bestellnummer 6672 002-0000	E-Adresse	A-Adresse 256., 259	Diag 2043*	K	Network Components Sensors Weiter FELDGERÄTE SIMATIC 300 SIMATIC 400 SIMATIC 400 SIMATIC HI Station SIMATIC PC Station SIMATIC PC Station
Einfügen mödlich						GGT2 002-0JD00 SIEMENS Kommunikationsmodul RF180C (IP67) zur Verbindung zweier Reader mit ₩

15. Doppelklicken Sie auf die eingefügte Baugruppe und ändern Sie den Gerätenamen auf RF180Cx110 und die IP-Adresse auf 192.168.0.110

igenschaften - RF18	юс							
Allgemein 10-Zyklus								
Kurzbezeichnung:	RF180C							
	Kommunikationsmodul RF180C (IP67) zur Verbindung zweier Reader mit PROFINET IO							
Bestell-Nr. / Firmware:	6GT2 002-0JD00							
Familie:	SIMATIC RFID							
Gerätename:	RF180Cx110							
Teilnehmer PROFINE	T IO-System							
Gerätenummer:	1 💌	PROFINET-IO-System (100)						
IP-Adresse:	192.168.0.110	Ethernet						
IP-Adresse durch	10-Controller zuweisen							
K								
Kommental.								
				~				
UK			Abbrechen	Hilte				

16. Die Hardware Konfiguration wird nun durch einen Klick auf , 💷 gespeichert und übersetzt.

5.3 Gerätenamen zuweisen



17. Markieren Sie die Baugruppe RF180C und wählen Sie anschließend über das Menü "Zielsystem" bei Ethernet **Gerätenamen vergeben**.

uta I	🖳 🖽 Konfig - [SIMATIC 300(1) (Konfiguration) RFID_RF180C]													
00	Station E	Bearbeiten	Einfüger	n Zie	elsystem	Ansicht Extras	Fenster Hilfe	9						_ 8 ×
0	🖻 🔓	H F 10	6	3	Laden ir Laden ir	n Baugruppe n PG		Ctrl+L						
	(0) U	R	07 5A		Baugrup Baugrup	ppen-Identifikation ppen-Identifikation	laden laden in PG		00)			S <u>u</u> chen:	<u></u>	nt ni
	2 1 CPU 315F-2 P X1 MPI/DP X2 X2 PN+0x100 X2/P1 Point 1 3 D116xDC24V 4 D116xDC24V			Gestörte Baugruppen Baugruppenzustand Betriebszustand Urlöschen Uhrzeit stellen		Ctrl+D Ctrl+I	-			Profil:	Standard PROFIBUS-DP PROFIBUS-PA PROFINET IO Drives Gateway			
٤ [6 7 8				Firmwar Geräter Etherne	e aktualisieren amen auf Memory t	Card speichern.		Ethernet-Te	ilnehmer l	v		HMI I/O Ident Systems ⊡ SIMATIC RFID 	
•	■ ⇒ [0:) RF1800	×110		PROFIB	US		۰,	Gerätename Gerätename	en überprü en vergeb	ifen en		 Network Components Sensors Weitere FELDGERÄTE 	
S	teckplatz 7 7	E BF	augruppe 1800x110 achtuse zwe	aier R	Serviced Reader	Jaten speichern Bestellnummer 6572 002 01000		E-Adresse	A-Adresse 256. 259	Diag 2003**	K	GGT2 00 SIEMEN zur Verbi	IMATIC 300 IMATIC 300 IMATIC 400 IIMATIC HMI Station IIMATIC PC Based Control IIMATIC PC Station 2-00D00 S ikationsmodul RF180C (IPE dung zweier Reader mit	300/400
, Verga	abe von PF	ROFINET I	O Device-0	ieräte	enamen.							0		Änd



Hinweis

Voraussetzung hierfür ist, dass die PG/PC- Schnittstelle auf TCP/IP eingestellt ist und die Netzwerkkarte des PCs richtig konfiguriert ist. Z.B.: IP- Adresse 192.168.0.99, Subnetz 255.255.255.0 und Router- Adresse -.-.- (Siehe Modul E02!)

18. Markieren Sie die SIMATIC RFID Baugruppe und Klicken Sie anschließend auf die Schaltfläche "Name zuweisen". Fenster wieder schließen.

Gerätenamen vergeben								
Gerätename: RF180Cx110 Cerätetyp: SIMATIC RFID								
Vorhandene Geräte:								
IP-Adresse MAC-Adresse Gerätetyp Gerätename	Name zuweisen							
192.168.0.110 08-00-06-99-28-BA SIMATIC REID RE180Cx110	Teilnehmer-Blinktest							
	Dauer (Sekunden): 3 💌							
	Blinken ein Blinken aus							
nur Geräte gleichen Typs anzeigen 🔲 nur Geräte ohne Namen anzeigen								
Aktualisieren Exportieren								
Schließen	Hilfe							

So sieht jetzt die fertige Hardware Konfiguration mit den zugehörenden Adressen aus



19. Dann kann die Hardware Konfiguration durch einen Klick auf , $\overset{\textcircled{}}{\underline{m}}$, in die SPS geladen werden. Dabei sollte der Betriebsartenschalter an der CPU auf Stop stehen! (\rightarrow $\overset{\textcircled{}}{\underline{m}}$)



dressen	von:			Adress	bereich von:	0	bis:	2047		
.PU 315	F-Z PN/DP			Freie A	dressvergabe:	Ja				
				Baugn Steckj	uppenträger/ platz:	0/2	CPU-	Nr. 1		
lter: 🔽	Eingänge	🔽 Ausg	änge 🔽	Adresslücken						
Тур	Adr. von	Adr. bis	Baugruppe		TPA	DP	PN	B	S	IF
Ξ	0	1	DI16xDC24	/	OB1-PA	-	-	0	4	-
	2	255	Lücke		•	-	•	-	•	•
Ξ	256	259	Anschluss z	weier Reader	OB1-PA	-	100(1)	-	1	0
-	260	2042	Lücke					-		-
×	2043	2043	RF180Cx11	D		-	100(1)		0	0
×	2044	2044	PN-I0x100		-	-	-	0	2	2
*	2045	2045	Port 1		-	-	-	0	2	3
*	2046	2046	PN-IOx100		-	-	-	0	2	2
	2047	2047	MPI/DP		-	-	-	0	2	1
A	U	3	Lucke				-		-	-
4	4	5	D016xDC24	W70.5A	UB1-PA	-	-	U	5	-
4	6	255	Lucke		-		-			÷
4	256	259	Anschluss z	weier Header	UBT-PA	-	100(1)		1	U
4	260	2047	Lucke		-	-	•			

20. Hardware Konfiguration schließen.

5.4 UDT Bausteine und FB45 einfügen



Bibliothek "**RFID_FB45_UDT_Bausteine**" aus dem Vorlagenverzeichnis dearchivieren und öffnen. Deutschsprachige **UDT11**, **UDT21**, **UDT 111**, **UDT261**, **UDT271**, **UDT281** und **FB45** kopieren und in den Bausteinordner des Projekts einfügen. Bibliothek wieder schließen.

BRFID_RF180C C:\Program	n Files\Siemens\	Step7\s7proj\RFID	_R_5	
PRID_RF180C SIMATIC 300(1) G CPU 315F-2 PN/DP G S7-Programm(2) G Quellen G Bausteine	Systemdaten ⊕ UDT21 ⊕ UDT281	😑 OB1 🔁 UDT111	₽ FB45 ₽ UDT261	UDT11
RFID_FB45_UDT_Bauste	ine C:\Progran	n Files\Siemens\St	ep7\S7libs\RFID_f	B4 🗖 🗖 🔀
PFID_FB45_UDT_Baustei S7-Programm(1) Quellen Bausteine	ne UDT20 UDT20 UDT61 UDT74 UDT110 UDT121 UDT121 UDT134 UDT260 UDT271 UDT284	UDT10 UDT21 UDT64 UDT100 UDT111 UDT124 UDT124 UDT261 UDT274	UDT11 UDT24 UDT70 UDT101 UDT114 UDT130 UDT130 UDT264 UDT280	UDT14 UDT60 UDT71 UDT104 UDT120 UDT131 UDT144 UDT1270 UDT220 UDT221



Hinweis

Anstelle der Bibliothek **RFID_FB45_UDT_Bausteine** kann auch das Beispielprogramm MOBY FB45 dearchiviert werden. Die Datei **FB45_V1_3.zip** finden Sie auf der RFID Systems CD in dem Verzeichnis "Daten", Unterverzeichnis "FB45"

BRFID_RF180C C:\Program	n Files\Siemens\	Step7\s7proj\RFID_R	L_5	
BFID_RF180C SIMATIC 300(1) G SIMATIC 300(1) G S7.Programm(2) G Quellen G Bausteine	Systemdaten ⊕ UDT21 ⊕ UDT281	⊕ OB1 ⊕ UDT111	₽ FB45 ₽ UDT261	⊕ UDT11 ⊕ UDT271
B MOBY FB45 C:\Program	Files\Siemens\S	tep7\S7Proj\FB45_V	1_3	
HOBY FB45	 OB1 OB100 DB11 DB48 DB63 DB75 UD720 UD761 UD774 UD7110 UD7121 UD7134 UD7260 UD7271 UD7284 	 0880 08121 DB12 DB60 DB64 UDT10 UDT64 UDT101111 UDT124 UDT140 UDT261 UDT274 Status Channel 1 	 0882 08122 0845 0861 0873 UDT11 UDT24 UDT70 UDT101 UDT101 UDT141 UDT1264 UDT280 Status Channel 2 	 OB86 FB45 DB47 DB62 DB74 UDT14 UDT60 UDT71 UDT104 UDT104 UDT120 UDT131 UDT144 UDT270 UDT281

5.5 Datenbausteine erstellen



DB48 Schreib- bzw. Lesedaten

Hier werden die Quell-Daten eines Schreib-Auftrags zum Transponder abgelegt oder die Ziel-Daten beim Lese-Auftrag vom Transponder gespeichert.

Datenbaustein DB48 erstellen

Eigenschaften - Datent	paustein	X				
Allgemein - Teil 1 Allgeme	in - Teil 2 Aufrufe Attribute					
Name und Typ:	DB48 Global-DB 💌] [
Symbolischer Name:	S_L_DATEN					
Symbolkommentar:	hier werden die Daten des Schreib-Lese Auftrags eingetragen	-				
Erstellsprache:	DB					
Projektpfad:						
Speicherort des Projekts:	C:\Program Files\Siemens\Step7\s7proj\RFID_R_5					
Erstellt am:	Code Schnittstelle 06.12.2009 08:47:43					
Zuletzt geändert am:	06.12.2009 08:47:43 06.12.2009 08:47:43					
Kommentar:						
OK	Abbrechen Hilfe					

DB48 öffnen 1024 Byte anlegen

Unter Name "Daten" eingeben.

Bei Typ (mit rechter Maustaste) Zusammengesetzte Datentypen das ARRAY Datenformat auswählen.

1..1024 innerhalb der Klammern eingeben.

In der nächsten Zeile BYTE auswählen bzw. eingeben.

H	🍇 KOP/AW	L/FUP - [DB48 "S_L_DATEN"	RFID_RF180C\SIMATIC 30	0(1)\CPU 315F-21	PN/DP\\DB48]				
4	🕞 Datei Bearbeiten Einfügen Zielsystem Test Ansicht Extras Fenster Hilfe								
	🗅 🖨 🔐 🛃 👗 🖻 🏨 🗠 બ 🕼 🏜 🔁 🏪 🔐 !« »! 🔲 🖳 💦								
ſ	Adresse	Name	Тур	Anfangswert	Kommentar				
I	*0.0		STRUCT						
	*0.0 +0.0	Daten	STRUCT ARRAY[11024]						
	*0.0 +0.0 *1.0	Daten	STRUCT ARRAY[11024] BYTE						

DB48 speichern und schließen





DB49 MDS Status Informationen

Im DB49 werden die MDS-Status Informationen eingetragen. Die Informationen werden in einem Datenbaustein mit einer festgelegten Datenstruktur abgelegt. MDS-Status Lesen mit Sub Command 01nach UDT261 oder Sub Command 02 nach UDT271.

Datenbaustein DB49 erstellen

Eigenschaften - Datenbaustein 🛛 🔀							
Allgemein - Teil 1 Allgeme	in - Teil 2 Aufrufe .	Attribute					
Name und Typ:	DB49	Global-DB	•	~			
Symbolischer Name:	MDS_STATUS						
Symbolkommentar:	hier werden die MDS	-Status Inform	nationen abgelegt				
Erstellsprache:	DB						
Projektpfad:							
Speicherort des Projekts:	C:\Program Files\Siemens\Step7\s7proj\RFID_R_5						
Erstellt am:	Code 06.12.2009 09:09:26		Schnittstelle				
Zuletzt geändert am:	06.12.2009 09:09:26		06.12.2009 09:09:26				
Kommentar:							
ОК			Abbrechen	Hilfe			

DB49 öffnen und UDT261 und UDT271 einfügen

Adresse	Name	ame Typ A		Kommentar
0.0		STRUCT		
+0.0	MDS_Status0	"MOBY P MDS-Status 0_d"		hier ist der UDT261 abgelegt
+18.0	MDS_Status1	"MOBY P MDS-Status 1_d"		hier ist der UDT271 abgelegt
=36.0		END_STRUCT		

Datensicht des DB49

Adresse	Name	Тур	Anfangswert	Aktualwert	Kommentar
0.0	MDS_Status0.reserved0	BYTE	B#16#0	B#16#0	
1.0	MDS_Status0.status_info	BYTE	B#16#0	B#16#0	Nodus MDS-Status
2.0	MDS_Status0.UID[1].Byte_1_4	DWORD	DW#16#0	DW#16#0	MDS-Nummer (unique identifier)
6.0	MDS_Status0.UID[1].Byte_5_8	DWORD	DW#16#0	DW#16#0	
10.0	MDS_Status0.MDS_type	BYTE	B#16#0	B#16#0	MDS-Typ
11.0	MDS_Status0.Lock_state	BYTE	B#16#0	B#16#0	Schreibschutzstatus EEPROM
12.0	MDS_Status0.reserved1[1]	BYTE	B#16#0	B#16#0	
13.0	MDS_Status0.reserved1[2]	BYTE	B#16#0	B#16#0	
14.0	MDS_Status0.reserved1[3]	BYTE	B#16#0	B#16#0	
15.0	MDS_Status0.reserved1[4]	BYTE	B#16#0	B#16#0	
16.0	MDS_Status0.reserved1[5]	BYTE	B#16#0	B#16#0	
17.0	MDS_Status0.reserved1[6]	BYTE	B#16#0	B#16#0	
18.0	MDS_Status1.reserved0	BYTE	B#16#0	B#16#0	
19.0	MDS_Status1.status_info	BYTE	B#16#0	B#16#0	Modus MDS-Status
20.0	MDS_Status1.UID[1].Byte_1_4	DWORD	DW#16#0	DW#16#0	MDS-Nummer (unique identifier)
24.0	MDS_Status1.UID[1].Byte_5_8	DWORD	DW#16#0	DW#16#0	
28.0	MDS_Status1.LFD	BYTE	B#16#0	B#16#0	Leistungsflussdichte: Beziehung zwischen Grenzwert und gemessenem Wert
29.0	MDS_Status1.FZP	BYTE	B#16#0	B#16#0	Fehlerzähler passiv (Ruhefehlerzähler)
30.0	MDS_Status1.FZA	BYTE	B#16#0	B#16#0	Fehlerzähler aktiv (Fehler während Kommunikation)
31.0	MDS_Status1.ANWZ	BYTE	B#16#0	B#16#0	Anwesenheitszähler
32.0	MDS_Statusl.reserved1[1]	BYTE	B#16#0	B#16#0	
33.0	MDS_Status1.reserved1[2]	BYTE	B#16#0	B#16#0	
34.0	MDS_Status1.reserved1[3]	BYTE	B#16#0	B#16#0	

Die Startadresse im DB49 ist bei Sub_Command 01 "0" und bei Sub_Command 02 "18" Die Datenlänge ist in beiden Fällen 18 Byte.

DB49 speichern und schließen





DB50 SLG-Status Informationen

Im DB50 werden die SLG-Status Informationen eingetragen. Je nach Auftrag müssen die Informationen in einem Datenbaustein mit einer festgelegten Datenstruktur abgelegt werden.

SLG-Status Lesen mit Sub_Command 01nach UDT111 oder Sub_Command 06 nach UDT281.

Datenbaustein DB50 erstellen

igenschaften - Datenbaustein 🛛 🔀								
Allgemein - Teil 1 Allgeme	in - Teil 2 Aufrufe /	Attribute						
Name und Typ:	DB50	Global-DB	•	-				
Symbolischer Name:	SLG_STATUS							
Symbolkommentar:	hier werden die SLG	Status Informat	iionen abgelegt					
Erstellsprache:	DB							
Projektpfad:								
Speicherort des Projekts:	C:\Program Files\Siemens\Step7\s7proj\RFID_R_5							
Erstellt am:	Code 06.12.2009 12:18:52		Schnittstelle					
Zuletzt geändert am:	06.12.2009 12:18:52		06.12.2009 12:18:52					
Kommentar:								
OK			Abbrechen	Hilfe				

DB50 öffnen und UDT111 und UDT281 einfügen

Öffnen Sie den DB50 und fügen Sie zu SLG_Status den UDT111ein. Fügen Sie zu SLG_Diagnose den UDT281 ein.

Adresse	Name Typ A		Anfangswert	Kommentar
0.0		STRUCT		
+0.0	SLG_Status	"MOBY SLG-Status_d"		hier ist der UDT111 abgelegt
+28.0	SLG_Diagnose	"MOBY P SLG-Status_d"		hier ist der UDT281 abgelegt
=56.0		END_STRUCT		



Datensicht des SLG-Status im DB50 (UDT111) Byte 0 bis 27

Adresse	Name	Тур	Anfangswert	Aktualwert	Kommentar
0.0	SLG_Status.status_info	BYTE	B#16#0	B#16#0	Modus SLG-Status
1.0	SLG_Status.hardware	CHAR	1.1	1.1	HW-Variante
2.0	SLG_Status.hardware_version	WORD	W#16#0	W#16#0	HW-Version
4.0	SLG_Status.loader_version	WORD	W#16#0	W#16#0	Urlader-Version
6.0	SLG_Status.firmware	CHAR	1.1	1.1	FW-Variante
8.0	SLG_Status.firmware_version	WORD	W#16#0	W#16#0	FW-Version
10.0	SLG_Status.driver	CHAR	1.1	1.1	Treiber-Variante
12.0	SLG_Status.driver_version	WORD	W#16#0	W#16#0	Treiber-Version
14.0	SLG_Status.interface	BYTE	B#16#0	B#16#0	Schnittstelle (RS 232 / RS 422)
15.0	SLG_Status.baud	BYTE	B#16#0	B#16#0	Baudrate
16.0	SLG_Status.reserved1	BYTE	B#16#0	B#16#0	Reserviert
17.0	SLG_Status.reserved2	BYTE	B#16#0	B#16#0	Reserviert
18.0	SLG_Status.reserved3	BYTE	B#16#0	B#16#0	Reserviert
19.0	SLG_Status.distance_limiting_SLG	BYTE	B#16#0	B#16#0	Reichweitenbegrenzung
20.0	SLG_Status.multitag_SLG	BYTE	B#16#0	B#16#0	Multitag SLG
21.0	SLG_Status.field_ON_control_SLG	BYTE	B#16#0	B#16#0	Berobetriebsart
22.0	SLG_Status.field_ON_time_SLG	BYTE	B#16#0	B#16#0	Berozeit
23.0	SLG_Status.sync_SLG	BYTE	B#16#0	B#16#0	Semaphorensteuerung (Synchronisation mit SLG)
24.0	SLG_Status.status_ant	BYTE	B#16#0	B#16#0	Status Antenne
25.0	SLG_Status.stand_by	BYTE	B#16#0	B#16#0	Standby Zeit nachdem ein Befehl ausgeführt wurde
26.0	SLG_Status.MDS_control	BYTE	B#16#0	B#16#0	Anvesenheit

Datensicht der SLG-Diagnose im DB50 (UDT281) Byte 28 bis 55

29.0	SLC Diemose status info	DVTP	P#16#0	P#16#0	Modue SIC-Status
20.0	SIG Diamaga R2D	DUTE	D#16#0	D#16#0	Pohlovnöhlov possiv (Dubofohlovnöhlov)
29.0	CLC Diagnose AP2	DUTE	D#10#0	D#16#0	Vehierzahler passiv (kunerenterzahler)
30.0	SLG_DIAGNOSE.ABZ	DIIL	D#10#0	D#10#0	Abbruchzanier
31.0	SLG_Diagnose.CFZ	BYTE	B#16#0	B#16#0	Codefehlerzähler
32.0	SLG_Diagnose.SFZ	BYTE	B#16#0	B#16#0	Signaturfehlerzähler
33.0	SLG_Diagnose.CRCFZ	BYTE	B#16#0	B#16#0	CRC-Fehlerzähler
34.0	SLG_Diagnose.BSTAT	BYTE	B#16#0	B#16#0	Befehlsstatus
35.0	SLG_Diagnose.ASMFZ	BYTE	B#16#0	B#16#0	Fehlerzähler Hostinterface (ASM)
36.0	SLG_Diagnose.reserved0[1]	BYTE	B#16#0	B#16#0	
37.0	SLG_Diagnose.reserved0[2]	BYTE	B#16#0	B#16#0	
38.0	SLG_Diagnose.reserved0[3]	BYTE	B#16#0	B#16#0	
39.0	SLG_Diagnose.reserved0[4]	BYTE	B#16#0	B#16#0	
40.0	SLG_Diagnose.reserved0[5]	BYTE	B#16#0	B#16#0	
41.0	SLG_Diagnose.reserved0[6]	BYTE	B#16#0	B#16#0	
42.0	SLG_Diagnose.reserved0[7]	BYTE	B#16#0	B#16#0	
43.0	SLC_Diagnose.reserved0[8]	BYTE	B#16#0	B#16#0	
44.0	SLG_Diagnose.reserved0[9]	BYTE	B#16#0	B#16#0	
45.0	SLG_Diagnose.reserved0[10]	BYTE	B#16#0	B#16#0	
46.0	SLG_Diagnose.reserved0[11]	BYTE	B#16#0	B#16#0	
47.0	SLG_Diagnose.reserved0[12]	BYTE	B#16#0	B#16#0	
48.0	SLG_Diagnose.reserved0[13]	BYTE	B#16#0	B#16#0	
49.0	SLG_Diagnose.reserved0[14]	BYTE	B#16#0	B#16#0	
50.0	SLG_Diagnose.reserved0[15]	BYTE	B#16#0	B#16#0	
51.0	SLG_Diagnose.reserved0[16]	BYTE	B#16#0	B#16#0	
52.0	SLG_Diagnose.reserved0[17]	BYTE	B#16#0	B#16#0	
53.0	SLG_Diagnose.reserved0[18]	BYTE	B#16#0	B#16#0	
54.0	SLG_Diagnose.reserved0[19]	BYTE	B#16#0	B#16#0	
55.0	SLG_Diagnose.reserved0[20]	BYTE	B#16#0	B#16#0	

Die Startadresse im DB50 ist bei Sub_Command 01 "**0**" SLG-Status Informationen und bei Sub_Command 06 "**28**" SLG-Diagnose Informationen Die Datenlänge ist in beiden Fällen 28 Byte.

DB50 speichern und schließen





DB47 Auftragsdatenbaustein

Der DB47 wird über den UDT11 im DB45 angesteuert. Im DB47 wird pro SLG (Schreib-Lese-Gerät) ein UDT21 eingebettet. Im jeweiligen UDT21 werden dann die Auftrags-Kommandos eingegeben. Die Daten des Schreib- bzw. Leseauftrags werden dann im DB48 abgelegt. Der Verweis auf den DB48 wird in den UDT21 des DB47 zugeordnet.

Datenbaustein DB47 erstellen

Eigenschaften - Datent	austein		X
Allgemein - Teil 1 Allgeme	in - Teil 2 Aufrufe Attri	bute	
Name und Typ:	DB47 GI	obal-DB	-
Symbolischer Name:	AUFTRAG		
Symbolkommentar:	hier werden die Kommar	ndos eingegeben	
Erstellsprache:	DB		
Projektpfad:			
Speicherort des Projekts:	C:\Program Files\Siemer	ns\Step7\s7proj\RFID_RF1	
Erstellt am:	Code 26.11.2009 13:18:30	Schnittstelle	
∠uletzt geandert am:	26.11.2009.13:18:30	26.11.2009.13:18:30	
Kommentar:			<u>~</u>
OK		Abbrechen	Hilfe

DB47 öffnen und pro Auftrag den UDT21 einfügen

DB47 öffnen und für SLG1 über ARRAY Format fünf Mal den UDT21 einfügen. Für SLG2 ebenfalls über ARRAY Format fünf Mal den UDT21 einfügen.

Adresse	Name	Тур	Anfangswert	Kommentar
*0.0		STRUCT		
+0.0	Kanal_l_Befehl	ARRAY[15]		SLG1 Aufträge 1 bis 5
*10.0		"MOBY CMD_d"		
+50.0	Kanal_2_Befehl	ARRAY[15]		SLG2 Aufträge 1 bis 5
*10.0		"MOBY CMD_d"		
=100.0		END_STRUCT		



Hinweis

In unserem Beispiel sind pro SLG bzw. Kanal fünf Auftragsbefehle möglich. Der UDT21 ist somit 10 Mal in den DB47 eingebettet. Die Befehle bzw. Aufträge des zweiten SLGs beginnen ab der Adresse 50.



Datensicht des DB47

Deca.INH		TTP.	ATAMASINPTIN	NUCLEAR	
0.0	Kanal_l_Befehl[1].command	BYTE	B#16#2	B#16#2	MDS Befehl:l=schreiben,2=lesen,3=init,4=slg-status,8=end,Å=set-ant,B=mds-status
1.0	Kanal_l_Befehl[1].sub_command	BYTE	B#16#0	B#16#0	Bitmuster für INIT-Befehl; Modus für END,SET-ANT,MDS-STATUS,SLG-STATUS
2.0	Kanal_l_Befehl[1].length	INI	Т	1	Anzahl der zu schreibenden/lesenden Daten in Bytes
4.0	Kanal_l_Befehl[1].address_MDS	WORD	M#16#0	N#16#0	Anfangsadresse auf MDS; Endadresse bei INIT;EW/Jahr für MDS-STATUS
6.0	Kanal_l_Befehl[1].DAT_DB_number	IMI	48	48	Nummer des DAT DB; Daten DB für NDS Daten
8.0	Kanal_1_Befehl[1].DAT_DB_address	IMI	0	0	Zeiger auf das Anfangswort im DAT DB
10.0	Kanal_1_Befehl[2].command	BYTE	B#16#2	B#16#2	MDS Befehl:l=schreiben,2=lesen,3=init,4=slg-status,8=end,A=set-ant,B=mds-status
11.0	Kanal_l_Befehl[2].sub_command	BYTE	B#16#0	B#16#0	Bitmuster für INIT-Befehl; Modus für END,SET-ANT,MDS-STATUS,SLG-STATUS
12.0	Kanal_1_Befehl[2].length	INT	L	т	Anzahl der zu schreibenden/lesenden Daten in Bytes
14.0	Kanal_l_Befehl[2].address_MDS	WORD	D#JE#0	D#JE#O	Anfangsadresse auf MDS; Endadresse bei INIT;KW/Jahr für MDS-STATUS
16.0	Kanal_l_Befehl[2].DAT_DB_number	INT	48	48	Nummer des DAT DB; Daten DB für NDS Daten
18.0	Kanal_l_Befehl[2].DAT_DB_address	INI	0	0	Zeiger auf das Anfangswort im DAT DB
20.0	Kanal_1_Befehl[3].command	BYTE	B#16#2	B#16#2	MDS Befehl:l=schreiben,2=lesen,3=init,4=slg-status,8=end,Å=set-ant,B=mds-status
21.0	Kanal_l_Befehl[3].sub_command	BYTE	B#16#0	B#16#0	Bitmuster für INIT-Befehl; Modus für END,SET-ANT,MDS-STATUS,SLG-STATUS
22.0	Kanal_l_Befehl[3].length	IMI	1	l	Anzahl der zu schreibenden/lesenden Daten in Bytes
24.0	Kanal_l_Befehl[3].address_MDS	WORD	M#16#0	N#16#0	Anfangsadresse auf MDS; Endadresse bei INIT;EW/Jahr für MDS-STATUS
26.0	Kanal_l_Befehl[3].DAT_DB_number	INT	48	48	Nummer des DAT DB; Daten DB für MDS Daten
28.0	Kanal_1_Befehl[3].DAT_DB_address	INI	0	0	Zeiger auf das Anfangswort im DAT DB
30.0	Kanal_l_Befehl[4].command	BYTE	B#16#2	B#16#2	MDS Befehl:l=schreiben,2=lesen,3=init,4=slg-status,8=end,A=set-ant,B=mds-status
31.0	Kanal_l_Befehl[4].sub_command	BYTE	B#16#0	B#16#0	Bitmuster für INIT-Befehl; Modus für END,SET-ANT,MDS-STATUS,SLG-STATUS
32.0	Kanal_l_Befehl[4].length	INT	1	T	Anzahl der zu schreibenden/lesenden Daten in Bytes
34.0	Kanal_l_Befehl[4].address_MDS	WORD	W#16#0	W#16#0	Anfangsadresse auf MDS; Endadresse bei INIT;KW/Jahr für MDS-STATUS
36.0	Kanal_l_Befehl[4].DAT_DB_number	INT	48	48	Nummer des DAT DB; Daten DB für NDS Daten
38.0	Kanal_l_Befehl[4].DAT_DB_address	INT	0	0	Zeiger auf das Anfangswort im DAT DB
40.0	Kanal_1_Befehl[5].command	BYTE	B#16#2	B#16#2	MDS Befehl:l=schreiben,2=lesen,3=init,4=slg-status,8=end,A=set-ant,B=mds-status
41.0	Kanal_1_Befehl[5].sub_command	BYTE	B#16#0	B#16#0	Bitmuster für INIT-Befehl; Modus für END,SET-ANT,MDS-STATUS,SLG-STATUS
42.0	Kanal_l_Befehl[5].length	INI	1	1	Anzahl der zu schreibenden/lesenden Daten in Bytes
44.0	Kanal_l_Befehl[5].address_MDS	WORD	W#16#0	W#16#0	Anfangsadresse auf MDS; Endadresse bei INIT;KW/Jahr für MDS-STATUS
46.0	Kanal_l_Befehl[5].DAT_DB_number	INT	48	48	Nummer des DAT DB; Daten DB für NDS Daten
48.0	Kanal_l_Befehl[5].DAT_DB_address	INI	0	0	Zeiger auf das Anfangswort im DAT DB
50.0	Kanal_2_Befehl[1].command	BYTE	B#16#2	B#16#2	MDS Befehl:l=schreiben,2=lesen,3=init,4=slg-status,8=end,Å=set-ant,B=mds-status
51.0	Kanal_2_Befehl[1].sub_command	BYTE	B#16#0	B#16#0	Bitmuster für INIT-Befehl; Modus für END,SET-ANT,MDS-STATUS,SLG-STATUS
52.0	Kanal_2_Befehl[1].length	INI	1	1	Anzahl der zu schreibenden/lesenden Daten in Bytes
54.0	Kanal_2_Befehl[1].address_MDS	WORD	W#16#0	W#16#0	Anfangsadresse auf MDS; Endadresse bei INIT;KW/Jahr für MDS-STATUS
56.0	Kanal_2_Befehl[1].DAT_DB_number	INT	48	48	Nummer des DAT DB; Daten DB für MDS Daten
58.0	Kanal_2_Befehl[1].DAT_DB_address	INT	0	0	Zeiger auf das Anfangswort im DAT DB
60.0	Kanal_2_Befehl[2].command	BYTE	B#16#2	B#16#2	MDS Befehl:l=schreiben,2=lesen,3=init,4=slg-status,8=end,Å=set-ant,B=mds-status
61.0	Kanal_2_Befehl[2].sub_command	BYTE	B#16#0	B#16#0	Bitmuster für INIT-Befehl; Modus für END,SET-ANT,MDS-STATUS,SLG-STATUS

(Ab Byte 50 beginnen die Befehle zu SLG2 bzw. Kanal 2)

DB47 speichern und schließen

DB45 SLG Parameter Baustein

Jeder MOBY-Kanal (SLG) benötigt seine eigenen Parameter. Diese sind in einer Datenstruktur als UDT 10 (mit englischen Kommentaren), UDT 11 (mit deutschen Kommentaren) bzw. UDT 14 (mit spanischen Kommentaren) vordefiniert. Sie müssen für jeden MOBY-Kanal diesen UDT in einem Datenbaustein aufrufen. Im UDT 11 sind unterschiedliche Variablen definiert:

• INPUT-Parameter: Diese Variablen müssen vom Anwender einmalig bei der Projektierung eingegeben werden (Ausnahme: command_DB_number / command_DB_address). Während der gesamten Laufzeit ist es nicht notwendig, diese Parameter zu verändern oder abzufragen. Bitte beachten Sie, dass nach Änderung eines INPUT-Parameters ein init_run durchzuführen ist, bevor die neue Einstellung wirksam wird (siehe Kapitel "Programmierung von Neustart und Wiederanlauf").

• Steuerbits: Mit diesen Boolschen Variablen startet der Anwender seine Befehle.

• Anzeigen: Die Anzeigen zeigen dem Anwender den Fortgang eines Befehls an. Fehleranalysen können einfach durchgeführt werden.

• FB-interne Variablen: Diese Variablen sind für den Anwender nicht von Bedeutung. Sie dürfen nicht durch die Anwendung verändert werden. Fehlfunktionen und Datenverfälschungen wären sonst die Folge.

	F
M	

1

Datenbaustein DB45 erstellen

Eigenschaften - Datenl	austein 🛛 🛛 🔀		
Allgemein - Teil1 Allgeme	n - Teil 2 Aufrufe Attribute		
Name:	DB45		
Symbolischer Name:	SLG_PARAMETER		
Symbolkommentar:	hier findet man die Parameter zu den SLGs		
Erstellsprache:	DB		
Projektpfad: BFID_RF180C\SIMATIC 300(1)\CPU 315F-2 PN/DP\S7-Programm (2)\Bausteine\DB45			
Speicherort des Projekts:	C:\Program Files\Siemens\Step7\s7proj\RFID_R_5		
Erstellt am:	Code Schnittstelle 06.12.2009 15:42:30		
Zuletzt geändert am:	06.12.2009 15:45:20 06.12.2009 15:45:20		
Kommentar:			
ОК	Abbrechen Hilfe		

DB45 öffnen und pro SLG den UDT11 einfügen

Im DB45 wird pro SLG der UDT11 aufgerufen.

Adresse	Name	Тур	Anfangswert	Kommentar
0.0		STRUCT		
+0.0	SLG_1	"MOBY Param_d"		Parameter des ersten SLG
+50.0	SLG_2	"MOBY Param_d"		Parameter des zweiten SLG
=100.0		END_STRUCT		





Eingaben im DB45

Die Basisadresse des RF180C ist 256 (siehe Hardware) hier bei Adresse 0.0 und 50.0 einzutragen Die Auswahl des SLG1 ist hier unter Adresse 2.0 einzutragen. Die Auswahl des SLG2 ist hier unter Adresse 52.0 einzutragen.

Im DB45 wird auf den DB47 (Auftragsdatenbaustein) verwiesen.

Die Aufträge des ersten SLGs sind im DB47 ab DBB0 hier bei Adresse 4.0 und 6.0 einzutragen Die Aufträge des zweiten SLGs sind im DB47 ab DBB50 hier bei Adresse 54.0 und 56.0 einzutragen

Alle anderen Werte werden nicht verändert.

Achten Sie bei den Eingaben im DB45 darauf, dass die Verkettung der DBs richtig fortgeführt wird.

Datensicht des DB45 bis Byte 33

Adresse	Name	Тур	Anfangswert	Aktualwert	Kommentar	
0.	0 SLG_1.ASM_address	INT	256	256	Input: Basisadresse des ASM (zyklisches Wort)	
2.	0 SLG_1.ASM_channel	INT	1	1	Input: Nummer des Kanals (1 4)	
4.	0 SLG_1.command_DB_number	INT	47	47	Input: Befehlsdatenbaustein-Nummer	
6.	0 SLG_1.command_DB_address	INT	0	0	Input: Anfangsadresse der Daten im BEDB	
8.	0 SLG_1.MDS_control	BYTE	B#16#1	B#16#1	Input: Anwesenheitskontrolle und MDS-Steuerung (0, 1, 2)	
9.	0 SLG_1.ECC_mode	BOOL	FALSE	FALSE	Input: Betriebsart mit ECC	
9.	1 SLG_1.RESET_long	BOOL	TRUE	TRUE	Input: true: langes RESET-Telegramm, nur für MOBY-Mode 5, 6	
10.	0 SLG_1.MOBY_mode	BYTE	B#16#5	B#16#5	Input: MOBY-Betriebsart	
11.	0 SLG_1.scanning_time	BYTE	B#16#0	B#16#0	Input: Abtastzeit für MOBY I/U	
12.	0 SLG_1.option_1	BYTE	B#16#0	B#16#0	Input: RESET-Befehl Option 1	
13.	0 SLG_1.distance_limiting	BYTE	B#16#F	B#16#F	Input: Reichweitenbegrenzung	
14.	0 SLG_1.multitag	BYTE	B#16#1	B#16#1	Input: max. Anzahl MDS im Feld	
15.	0 SLG_1.field_ON_control	BYTE	B#16#0	B#16#0	Input: BERO-Betriebsart	
16.	0 SLG_1.field_ON_time	BYTE	B#16#0	B#16#0	Input: BER0-Zeit	
17.	0 SLG_1.reserved	BYTE	B#16#0	B#16#0		
18.	0 SLG_1.ANZ_MDS_present	BOOL	FALSE	FALSE	Anwesenheit eines MDS	
18.	1 SLG_1.ANZ_cancel	BOOL	FALSE	FALSE	Cancel-Bit im PEW ist gesetzt	
18.	2 SLG_1.ANZ_ECC	BOOL	FALSE	FALSE	Fehlerkorrektur wurde durchgeführt	
18	3 SLG_1.reserved0	BOOL	FALSE	FALSE		
18.	4 SLG_1.LR_bat	BOOL	FALSE	FALSE	Batterie des MDS 507	
18	5 SLG_1.battery_low	BOOL	FALSE	FALSE	Batterieüberwachung	
18.	6 SLG_1.error	BOOL	FALSE	FALSE	Fehler während der Befehlsbearbeitung	
18	7 SLG_1.ready	BOOL	FALSE	FALSE	Der Befehl bzw. Befehlskette ist ausgeführt	
19.	0 SLG_1.cancel	BOOL	FALSE	FALSE	Set: Befehl bzw. Befehlskette abbrechen	
19.	1 SLG_1.command_start	BOOL	FALSE	FALSE	Set: Startsignal für Befehl bzw. Befehlskette	
19.	2 SLG_1.repeat_command	BOOL	FALSE	FALSE	Set: Letzten Befehl wiederholen	
19.	3 SLG_1.init_run	BOOL	TRUE	TRUE	Set: ASM rücksetzen und neu parametrieren	
19.	4 SLG_1.ASM_failure	BOOL	FALSE	FALSE	OB122 Set: ASM ausgefallen	
19.	5 SLG_1.FB45_active	BOOL	FALSE	FALSE	FC-aktiv	
19.	6 SLG_1.ANZ_next	BOOL	FALSE	FALSE	Befehl NEXT war letzter Befehl	
19.	7 SLG_1.ANZ_reset	BOOL	FALSE	FALSE	Befehl RESET war letzter Befehl	
20.	0 SLG 1.ASM busy	BOOL	FALSE	FALSE	ASM bearbeitet einen Befehl	
20.	1 SLG_1.command_rep_active	BOOL	FALSE	FALSE	ASM führt Befehlswiederholung durch	
21.	0 SLG 1.number MDS	BYTE	B#16#0	B#16#0	Anzahl der im Feld befindlichen MDS	
22.	0 SLG 1.error MOBY	BYTE	B#16#0	B#16#0	Fehleranzeige vom Anschaltmodul	
23.	0 SLG 1.error FB	BYTE	B#16#0	B#16#0	Fehleranzeige von FC	
24.	0 SLG 1.error BUS	WORD	W#16#0	W#16#0	Fehleranzeige von PROFIBUS	
26.	0 SLG_1.version_MOBY	WORD	W#16#0	W#16#0	Firmwareversion MOBY	
28.	0 SLG_1.counter_customer	BYTE	B#16#0	B#16#0	Interner Schleifenzähler	
29.	0 SLG_1.counter_notused	BYTE	B#16#0	B#16#0	FB intern	
30.	0 SLG_1.reserved1[1]	DWORD	DW#16#0	DW#16#0	FB-interne Variablen. Diese dürfen vom Anwender nicht verändert werden.	



Datensicht des DB45 ab Byte 34

34.0	SLG_1.reserved1[2]	DWORD	DW#16#0	DW#16#0	
38.0	SLG_1.reserved1[3]	DWORD	DW#16#0	DW#16#0	
42.0	SLG_1.reserved2	WORD	W#16#0	W#16#0	FB intern
44.0	SLG_1.initRUN_timeout	INT	1000	1000	FB intern
46.0	SLG_1.PEW_timout_ASM_fail	BYTE	B#16#5	B#16#5	FB intern
47.0	SLG_1.PEW_timeout	BYTE	B#16#32	B#16#32	FB intern
48.0	SLG_1.reserved3	BYTE	B#16#0	B#16#0	FB intern
49.0	SLG_1.Testbyte	BYTE	B#16#0	B#16#0	FB intern
50.0		INT	256	256	Input: Basisadresse des ASM (zyklisches Wort)
52.0	SLG 2.ASM channel	INT	1	2	Input: Nummer des Kanals (1 4)
54.0	SLG_2.command_DB_number	INT	47	47	Input: Befehlsdatenbaustein-Nummer
56.0	SLG_2.command_DB_address	INT	0	50	Input: Anfangsadresse der Daten im BEDB
58.0	SLG_2.MDS_control	BYTE	B#16#1	B#16#1	Input: Anwesenheitskontrolle und MDS-Steuerung (0, 1, 2)
59.0	SLG 2.ECC mode	BOOL	FALSE	FALSE	Input: Betriebsart mit ECC
59.1	SLG_2.RESET_long	BOOL	TRUE	TRUE	Input: true: langes RESET-Telegramm, nur für MOBY-Mode 5, 6
60.0	SLG 2.MOBY mode	BYTE	B#16#5	B#16#5	Input: MOBY-Betriebsart
61.0	SLG_2.scanning_time	BYTE	B#16#0	B#16#0	Input: Abtastzeit für MOBY I/U
62.0	SLG 2.option 1	BYTE	B#16#0	B#16#0	Input: RESET-Befehl Option 1
63.0	SLG 2.distance limiting	BYTE	B#16#F	B#16#F	Input: Reichweitenbegrenzung
64.0	SLG 2.multitag	BYTE	B#16#1	B#16#1	Input: max. Anzahl MDS im Feld
65.0	SLG_2.field_ON_control	BYTE	B#16#0	B#16#0	Input: BER0-Betriebsart
66.0	SLG 2.field ON time	BYTE	B#16#0	B#16#0	Input: BER0-Zeit
67.0	SLG 2.reserved	BYTE	B#16#0	B#16#0	
68.0	SLG 2.ANZ MDS present	BOOL	FALSE	FALSE	Anwesenheit eines MDS
68.1	SLG 2.ANZ cancel	BOOL	FALSE	FALSE	Cancel-Bit im PEW ist gesetzt
68.2	SLG 2.ANZ ECC	BOOL	FALSE	FALSE	Fehlerkorrektur wurde durchgeführt
68.3	SLG 2.reserved0	BOOL	FALSE	FALSE	-
68.4		BOOL	FALSE	FALSE	Batterie des MDS 507
68.5	SLG 2.battery low	BOOL	FALSE	FALSE	Batterieüberwachung
68.6	SLG 2.error	BOOL	FALSE	FALSE	Fehler während der Befehlsbearbeitung
68.7		BOOL	FALSE	FALSE	Der Befehl bzw. Befehlskette ist ausgeführt
69.0	SLG 2.cancel	BOOL	FALSE	FALSE	Set: Befehl bzw. Befehlskette abbrechen
69.1		BOOL	FALSE	FALSE	Set: Startsignal für Befehl bzw. Befehlskette
69.2	SLG 2.repeat command	BOOL	FALSE	FALSE	Set: Letzten Befehl wiederholen
69.3	SLG 2.init run	BOOL	TRUE	TRUE	Set: ASM rücksetzen und neu parametrieren
69.4	SLG 2.ASM failure	BOOL	FALSE	FALSE	OB122 Set: ASM ausgefallen
69.5	SLG 2.FB45 active	BOOL	FALSE	FALSE	FC-aktiv
69.6	SLG 2.ANZ next	BOOL	FALSE	FALSE	Befehl NEXT war letzter Befehl
69.7	SLG 2.ANZ reset	BOOL	FALSE	FALSE	Befehl RESET war letzter Befehl
70.0	SLG 2.ASM busy	BOOL	FALSE	FALSE	ASM bearbeitet einen Befehl
70.1	SLG_2.command_rep_active	BOOL	FALSE	FALSE	ASM führt Befehlswiederholung durch
71.0	SLG_2.number_MDS	BYTE	B#16#0	B#16#0	Anzahl der im Feld befindlichen MDS
72.0	SLG_2.error_MOBY	BYTE	B#16#0	B#16#0	Fehleranzeige vom Anschaltmodul
73.0	SLG_2.error_FB	BYTE	B#16#0	B#16#0	Fehleranzeige von FC
74.0	SLC_2.error_BUS	WORD	W#16#0	W#16#0	Fehleranzeige von PROFIBUS
76.0	SLG 2.version MOBY	WORD	W#16#0	W#16#0	Firmwareversion MOBY
78.0	SLG 2.counter customer	BYTE	B#16#0	B#16#0	Interner Schleifenzähler
79.0	SLG 2.counter notused	BYTE	B#16#0	B#16#0	FB intern
80.0	SLG 2.reserved1[1]	DWORD	DW#16#0	DW#16#0	FB-interne Variablen. Diese dürfen vom Anwender nicht verändert werden.
84.0	SLG 2.reserved1[2]	DWORD	DW#16#0	DW#16#0	
88.0	SLG 2. reserved1[3]	DWORD	DW#16#0	DW#16#0	
92 0	SLG 2. reserved2	WORD	W#16#0	W#16#0	FB intern
94 0	SLG 2. initRUN timeout	INT	1000	1000	FB intern
96.0	SLG 2. PEW timout ASM feil	BYTE	B#16#5	B#16#5	FB intern
97.0	SLG 2. PEW timeout.	BYTE	B#16#32	B#16#32	FB intern
98.0	SLG 2. reserved3	BYTE	B#16#0	B#16#0	FB intern
99.0	SLG 2. Testhyte	BYTE	B#16#0	B#16#0	FB intern
L			- / 20 / 0	- ******	

DB45 speichern und schließen



Hinweis

Die Eingaben im Datenbaustein werden hier als Aktualwert eingetragen. Achten Sie darauf, dass beim Initialisieren des Datenbausteines oder Urlöschen der CPU die Aktualwerte mit den Anfangswerten des Datenbausteines überschrieben werden.

Weitere Informationen zum Aufbau der Datenstruktur mit Datenbausteinen und zu den einzelnen UDTs finden Sie im Funktionshandbuch "RFID Systems FB45".

5.6 Programmierung von Neustart und Wiederanlauf

1

Ein Neustart des MOBY-ASM (An-Schalt-Modul) erfolgt durch das Setzen der Variablen "**init_run**". Mit "**init_run**" werden das ASM und der FB 45 neu parametriert und synchronisiert.

Ein "init_run" ist notwendig nach

- dem Einschalten der SIMATIC (OB 100)
- dem Einschalten der Versorgungsspannung am ASM
- der Unterbrechung der PROFINET-Kommunikation
- einer Fehlermeldung durch die Variable "error_BUS"
- einem Transpondertypwechsel z.B. von RF300 auf ISO oder umgekehrt
- nach Änderung eines INPUT-Parameters im DB45 (SLG_Parameter)



OB100 erstellen

Eigenschaften - Organ	isationsbaustein 🛛 🛛 🕅
Allgemein - Teil 1 Allgeme	in - Teil 2 Aufrufe Attribute
Name:	08100
Symbolischer Name:	NEUSTART
Symbolkommentar:	Programmierung von Neustart und Wiederanlauf
Erstellsprache:	FUP
Projektpfad:	
Speicherort des Projekts:	C:\Program Files\Siemens\Step7\s7proj\RFID_RF1
Erstellt am:	Code Schnittstelle 26.11.2009 17:35:45
∠uletzt geandert am:	26.11.200917:35:45 26.11.200917:35:45
Kommentar:	
ОК	Abbrechen Hilfe

OB100 öffnen und Programm eingeben

OB100 : Neustart und Wiederanlauf

Kommentar:							
Netzwerk 1	: Setzen der Variablen "init_run"						
Kommentar:							
SET S S	"SLG_PARAMETER".SLG_1.init_run "SLG_PARAMETER".SLG_2.init_run	DB45.DBX19.3 DB45.DBX69.3	Set: Set:	ASM ASM	rücksetzen rücksetzen	und n und n	eu parametrieren eu parametrieren

OB100 speichern und schließen

5.7 FC11 Funktion für einen Befehl bzw. Auftrag

1

Bevor Sie einen MOBY-Befehl mit **"command_start**" starten können, müssen Sie diesen definieren. Für die einfache Definition eines Befehls steht Ihnen der der UDT 21 (deutsche Kommentare) zur Verfügung. Der UDT21 ist in den DB47 mehrfach eingebettet. Damit nicht alle Eingaben im Datenbaustein DB47 durchgeführt werden müssen, soll hier ein Baustein für einen Befehl bzw. Auftrag erstellt werden.

Dieser Baustein kann dann im Steuerungsprogramm z.B. bei einer Befehlskette mehrfach aufgerufen werden.



Funktion FC11 erstellen

Legen Sie eine neue Funktion FC11 an.

igenschaften - Funktion 🛛 🔀						
Allgemein - Teil1 Allgeme	in - Teil 2 Aufrufe Attribute					
Name:	FC11					
Symbolischer Name:	AUFTRAG_BEFEHL					
Symbolkommentar:	Baustein für einen Auftrag bzw. B	efehl				
Erstellsprache:	FUP					
Projektpfad:	RFID_IEC_Teil2\SIMATIC 300(1)" Programm(1)\Bausteine\FC11	CPU 315F-2 PN/DP\S7-				
Speicherort des Projekts:	C:\Program Files\Siemens\Step7\	s7proj\RFID_I_1				
Erstellt am:	Code 03.12.2009 18:58:48	Schnittstelle				
Zuletzt geändert am:	03.12.2009 17:46:22	03.12.2009 17:46:22				
Kommentar:						
	1					
OK		Abbrechen Hilfe				

IN Variablen anlegen

	Inhalt von: 'Umgebung\Schnittstelle\IN'								
🖃 🕕 Schnittstelle	^	Name	Datentyp	Kommentar					
		🖼 slg_command	Byte						
🔤 slg_command		😉 slg_sub_command	Byte						
🖾 🖾 slg_sub_command		🛅 in_length	Int						
🖾 in_length	≡	🛅 in_mds_ADDR	Int						
🖾 in_mds_ADDR		🛅 in_dat_DB_Nr	Int						
🗂 in_dat_DB_Nr		🔚 in_dat_DB_ADDR	Int						
In_dat_DB_ADDR									

OUT Variablen anlegen

	Inhalt von: 'Umgebung\Schnittstelle\OUT'					
	~		Name	Datentyp	Kommentar	
⊨ -⊒- оот		12	command	Byte		
E conmand		12	sub_command	Byte		
🔤 sub_command		12	out_length	Int		
🔤 out_length	≣	12	out_mds_ADDR	Word		
🗠 🖾 out_nds_ADDR		12	out_dat_DB_Nr	Int		
🖂 🖾 out_dat_DB_Nr		12	out_dat_DB_ADDR	Int		
□ Out_dat_DB_ADDR		Ð				



Netzwerke 1 bis 6 eingeben

FC11 : Auftrag bzw. Befehl

Netzwerk 1 : Kommando







Netzwerk 3 : Datenlänge



Netzwerk 4 : MDS-Startadresse



Netzwerk 5: DB-Nr. für Datenablage Ziel bzw. Quelle



Netzwerk 6: Ziel- bzw. Quellstartadresse des DBs der Datenablage



FC11 speichern und schließen

5.8 Grundlagen zu den Eingaben am Kommando-Baustein FC11

1

Über die Eingangs-Variablen des FC11 werden die Werte für den DB47 vorgegeben. Pro Auftrag bzw. Befehl wird ein UDT21 verwendet.

Datensicht des UDT21

ſ	Adresse	Name	Тур	Anfangswert	Kommentar
ľ	0.0		STRUCT		
	+0.0	command	BYTE	B#16#2	MDS Befehl:l=schreiben,2=lesen,3=init,4=slg-status,8=end,A=set-ant,B=mds-status
	+1.0	sub_command	BYTE	B#16#0	Bitmuster für INIT-Befehl; Modus für END,SET-ANT,MDS-STATUS,SLG-STATUS
	+2.0	length	INT	1	Anzahl der zu schreibenden/lesenden Daten in Bytes
	+4.0	address_MDS	WORD	W#16#0	Anfangsadresse auf MDS; Endadresse bei INIT;KW/Jahr für MDS-STATUS
	+6.0	DAT_DB_number	INT	48	Nummer des DAT DB; Daten DB für MDS Daten
I	+8.0	DAT_DB_address	INT	0	Zeiger auf das Anfangswort im DAT DB
1	=10.0		END_ST:		

Befehlsübersicht

Command [hex]		Befehl				
normal	gekettet *					
01	41	Daten auf MDS schreiben				
02	42	Daten vom MDS lesen				
03	43	MDS initialisieren				
04	44	SLG-Status				
06	_	NEXT				
08	48	END; Kommunikation mit dem MDS beenden				
0A	4A	Antenne Ein/Aus				
0B	4B	MDS-Status				

*) Gekettete Befehle werden nicht von allen SLG bzw. ASM unterstützt. Beachten Sie hierzu die Angaben in den MOBY-Handbüchern für Projektierung, Montage und Service.

Daten auf Transponder schreiben

Command	sub_command	length	address_MDS	DAT_DB	Bemerkung
[hex]	[hex]	[dez]	[hex]	[dez]	
01	_	1 bis 32767 * Länge der zu schreibenden MDS-Daten	0000 bis FFFF Ab dieser Anfangsadresse werden die Daten auf den MDS geschrieben.	Zeiger auf die Nutzdaten, die auf den MDS geschrieben werden sollen	

Daten vom Transponder lesen

Command	sub_command	length	address_MDS	DAT_DB	Bemerkung
[hex]	[hex]	[dez]	[hex]	[dez]	
02	-	1 bis 32767 * Länge der zu lesenden MDS-Daten	0000 bis FFFF Ab dieser Anfangs- adresse werden die Daten vom MDS gelesen.	Zeiger auf die Nutzdaten. Der FB 45 hinterlegt hier die gelesenen MDS-Daten	



Transponder initialisieren

Command	sub_command	length	address_MDS	DAT_DB	Bemerkung
[hex]	[hex]	[dez]	[hex]	[dez]	
03	00 bis FF Hexwert, mit dem der MDS beschrieben wird	-	Speichergröße des zu initialisierenden MDS	-	

SLG-Status auslesen

Command [hex]	sub_command [hex]	length [dez]	address_MDS [hex]	DAT_DB [dez]	Bemerkung
04	01 = Status nach UDT 110 ¹	-	-	Zeiger auf Ergebnis.	MOBY U/D bzw. RF300
	02 = Status nach UDT 120 ¹ (letzte Befehle)			Das Ergebnis wird mit dem entsprechenden	MOBY U
	03 = Status nach UDT 130 ¹			UDT dargestellt (siehe	MOBY U
	(Fehlermeldungen) 04 = Status nach UDT 140 ¹			sub_command)	MOBY U
	(MDS im Feld) 05 = Status nach UDT 150				MOBY U
	(Kommunikationsgüte) 06 = Status nach UDT 280				RF300
	(Diagnosedaten)				

(Wir verwenden in unserem Beispiel die deutschsprachigen UDT111 und UDT281 zu RF300)

Befehl NEXT

Command	sub_command	length	address_MDS	DAT_DB	Bemerkung
[hex]	[hex]	[dez]	[hex]	[dez]	
06	-	-	-	-	NEXT: die Bearbeitung dieses MDS ist abgeschlossen

Befehl END

Command [hex]	sub_command [hex]	length [dez]	address_MDS [hex]	DAT_DB [dez]	Bemerkung
08	00 = Die Bearbeitung mit dem MDS ist beendet	-	-	-	ANZ_MDS_present wird zurückgesetzt
	01 = Bearbeitungs- pause mit dem MDS ¹				ANZ_MDS_present bleibt gesetzt

1

Antenne des SLG ein bzw. ausschalten

Command	sub_command	length	address_MDS	DAT_DB	Bemerkung
[hex]	[hex]	[dez]	[hex]	[dez]	
0A	01 = Antenne einschalten 02 = Stand-by; Antenne ausschalten 09 = Antenne auf die Umgebung abstimmen (FFT)	-	_	-	Der Befehl Antenne Ein/Aus kann nicht mit Befehlswiederholung (siehe Kapitel "Befehlswiederholung") gestartet werden. nur SLG 80 (MOBY F)

Transponder Status

Command [hex]	sub_command [hex]	length [dez]	address_MDS [hex]	DAT_DB [dez]	Bemerkung
0B	00 = Status und Diagnose	-	Heutiges Datum (Woche/Jahr) zur Berech- nung der Batterie- lebensdauer (z. B. 1401 hex = 20. Woche im Jahr 2001)	Zeiger auf Ergebnis. Das Ergebnis wird mit UDT 100 dargestellt.	nur MOBY U (siehe Kapitel "Die UDTs des FB 45")
	01 = Typ und Schreibschutzstatus	-	-	Zeiger auf Ergebnis. Das Ergebnis wird mit UDT 260 dargestellt.	RF300 (siehe Kapitel "Die UDTs des FB 45")
	02 = Diagnosedaten	-	_	Zeiger auf Ergebnis. Das Ergebnis wird mit UDT 270 dargestellt.	RF300 (siehe Kapitel "Die UDTs des FB 45")

(Wir verwenden in unserem Beispiel die deutschsprachigen UDT261 und UDT271 zu RF300)

Notizen

5.9 Befehlskette

1

Der Aufbau einer Befehlskette erfolgt durch den Anwender, indem entsprechend viele UDT 21 hintereinander in einem DB abgelegt werden. Dabei müssen aneinander gekettete Befehle alle vom "Command"-Typ "4x" sein. Der letzte Befehl in einer Kette muss vom Typ 0x sein. Daran erkennt der FB 45 das Ende einer Befehlskette.

Beispiel für eine Befehlskette im DB47

Von einem MDS sollen 4 Datensätze bearbeitet werden. Die Befehlsstruktur ist wie unten dargestellt im Auftrags-DB hinterlegt. Die Ziel und Quell-Daten des MDS werden im DB 48 fortlaufend abgelegt.

Lese MDS-Adresse 0000 hex Länge 600 Lese MDS-Adresse 1000 hex Länge 100 Lese MDS-Adresse 1200 hex Länge 1 Schreibe MDS-Adresse 1200 hex Länge 1

Name	Anfangswert	Kommentar
Teilbefehl [1].command	42	Lesebefehl; es folgt ein weiterer Befehl
Teilbefehl [1].pattern	00	
Teilbefehl [1].length	600	
Teilbefehl [1].adress_MDS	0000	
Teilbefehl [1].DAT_DB_number	48	
Teilbefehl [1].DAT_DB_address	0	
Teilbefehl [2].command	42	Lesebefehl; es folgt ein weiterer Befehl
Teilbefehl [2].pattern	00	
Teilbefehl [2].length	100	
Teilbefehl [2].adress_MDS	1000	
Teilbefehl [2].DAT_DB_number	48	
Teilbefehl [2].DAT_DB_address	600	
Teilbefehl [3].command	42	Lesebefehl; es folgt ein weiterer Befehl
Teilbefehl [3].pattern	00	
Teilbefehl [3].length	1	
Teilbefehl [3].adress_MDS	1200	
Teilbefehl [3].DAT_DB_number	48	
Teilbefehl [3].DAT_DB_address	700	
Teilbefehl [4].command	01	Schreibbefehl; letzter Befehl in der Kettung
Teilbefehl [4].pattern	00	
Teilbefehl [4].length	1	
Teilbefehl [4].adress_MDS	1200	
Teilbefehl [4].DAT_DB_number	48	
Teilbefehl [4].DAT_DB_address	701	

Notizen

5.10 Grundlagen zum FB45 und DB45

i

Der FB45 greift auf den DB45 zu. Im DB45 ist pro SLG ein UDT11 eingebettet.

Datensicht des UDT11

Adresse	Name	Тур	Anfangswert	Kommentar
0.0		STRUCT		
10.0	JCM address	тып	0	Input, Designdresse des ACM
+0.0	ASM_address	INT	0	(zvklisches Wort)
+2.0	ASM channel	INT	1	Input: Nummer des Kanals (1 4)
+4.0	command DB number	INT	47	Input: Befehlsdatenbaustein-Nummer
+6.0	command_DB_address	INT *	0	Input: Anfangsadresse der Daten im
+8.0	MDS_control	BYTE	B#16#1	Input: Anwesenheitskontrolle und
+9 0	FCC mode	BOOT.	FALSE	Input: Betriebsart mit ECC
+9 1	BESET long	BOOL	FALSE	Input: true: langes RESET-
19.1	TEPET_TONG	DOOT	11101	Telegramm, nur für MOBY-Mode 5
+10.0	MOBY mode	BYTE	B#16#1	Input: MOBY-Betriebsart
+11.0	scanning time	BYTE	B#16#0	Input: Abtastzeit für MOBY I/U
+12.0	option 1	BYTE	B#16#0	Input: RESET-Befehl Option 1
+13.0	distance_limiting	BYTE	B#16#F	Input: Reichweiten- /Leistungseinstellung
+14.0	multitag	BYTE	B#16#1	Input: max. Anzahl MDS im Feld
+15.0	field ON control	BYTE	B#16#0	Input: BERO
+16.0	field ON time	BYTE	B#16#0	Input: MOBY U: BERO-Zeit
+17.0	no convodů	DVME	D#16#0	MOBY D: MDS-Typ
+17.0	Teservedo	DITE	D#10#0	
+18.0	ANZ MDS present	BOOL	FALSE	Anwesenheit eines MDSs
+18.1	ANZ cancel	BOOL	FALSE	Cancel-Bit im PEW ist gesetzt
+18.2	ANZ_ECC	BOOL	FALSE	Fehlerkorrektur wurde durchgeführt
+18.3	reserved	BOOL	FALSE	
±18 4	IR hat	BOOT.	FALGE	Patterie des MDS 507
110.4	battern len	BOOL	PALOE	Datterie des MDS 507
+10.5	Dattery_10w	BOOL	PALSE	Batterreuberwachung
+18.6	error	ROOT	FALSE	Fenier Wanrend der Befehlsbearbeitung
+18.7	ready	BOOL	FALSE	Der Befehl bzw. Befehlskette ist ausgeführt
+19.0	cancel	BOOL	FALSE	Set: Befehl bzw. Befehlskette abbrechen
+19.1	command_start	BOOL	FALSE	Set: Startsignal für Befehl bzw. Befehlskette
+19.2	repeat_command	BOOL	FALSE	Set: letzten Befehl wiederholen
+19.3	init_run	BOOL	TRUE	Set: ASM rücksetzen und neu parametrieren
+19.4	ASM failure	BOOL	FALSE	OB122 Set: ASM ausgefallen
+19.5	FB45 active	BOOL	FALSE	FB-aktiv
+19.6	ANZ next	BOOL	FALSE	Befehl NEXT war letzter Befehl
+19 7	ANZ reset	BOOL	FALSE	Befehl RESET war letzter Befehl
+20.0	ASM buey	BOOL	FALSE	ASM hearbeitet einen Befehl
+20.1	command_rep_active	BOOL	FALSE	ASM führt Befehlswiederholung
+21.0	number_MDS	BYTE	B#16#0	Anzahl der im Feld befindlichen
	NARY		5/16/10	MDS
+22.0	error MOBI	BITE	B#16#0	Fenieranzeige vom Anschaltmodul
+23.0	error FB	BYTE	B#16#0	Fehleranzeige von FB
+24.0	error_BUS	WORD	W#16#0	Fehleranzeige von PROFIBUS
+26.0	version MOBY	WORD	W#16#0	Firmwareversion MOBY
+28.0	reserved2	ARRAY [14]	DW#16#0	FB-interne Variablen. Diese dürfen vom Anwender nicht verändert werden.
*4.0		DWORD		
+44.0	initRUN timeout	INT	1000	FB-interne Variablen. Diese dürfen
	-		- ULCUS	vom Anwender nicht verändert werden.
+46.0	PEW_timeout_ASM_fail	BALE	B#16#5	Vom Anwender nicht verändert werden.
+47.0	PEW_timeout	BYTE	B#16#32	FB-interne Variablen. Diese dürfen vom Anwender nicht verändert werden.
+48.0	reserved3	BYTE	B#16#0	FB-interne Variablen. Diese dürfen vom Anwender nicht verändert werden.
+49.0	Testbyte	BYTE	B#16#0	FB-interne Variablen. Diese dürfen vom Anwender nicht verändert werden.
=50.0		END_STRUCT		

i

Input-Parameter des UDT11

Diese Variablen müssen vom Anwender einmalig bei der Projektierung eingegeben werden (Ausnahme: command_DB_number / command_DB_address). Während der gesamten Laufzeit ist es nicht notwendig, diese Parameter zu verändern oder abzufragen.

Bitte beachten Sie, dass nach Änderung eines INPUT-Parameters ein "init_run" durchzuführen ist, bevor die neue Einstellung wirksam wird (siehe Kapitel "Programmierung von Neustart und Wiederanlauf").

Variable	Beschreibung						
ASM_address	Logische Basisadresse des ASM; diese Adresse muss mit der "Anfangsadresse" des ASM in HW- Konfig des SIMATIC-Managers übereinstimmen. Bitte beachten Sie, dass diese Adresse nichts mit der PROFIBUS-Adresse zu tun hat, welche am ASM bzw. ET 200M eingestellt wird.						
ASM_channel	Numme	er des MOBY-Kanals, mit dem gearbeitet w	erden soll:				
		ASM-Typ		Werteberei	ch		
	ASM 4 ASM 4 ASM 4	75, 452, 456; RF170C 54, 754, 854 73, 850	1, 2 1, 2, 3, 4 1				
command_DB_ number	Numme ist	er des Datenbausteins, in dem der MDS-Be	fehl festgelegt	Diese INPU immer verär	T-Parameter dürfen ndert werden, wenn		
command_DB_ address	Adress beginn "comm nächste	Adresse innerhalb des "command_DB". Auf dieser Adresse beginnt der nächste MDS -Befehl. "command_DB_number" und "command_DB_address" bilden einen Datenzeiger auf den nächsten Befehl (siehe Kapitel "Projektierungsschema").					
	Beachte: Die Input-Parameter command_DB_number und command_DB_:address dürfen nur verändert werden, wenn ready =1 ist. Nach einer Änderung dieser Parameter braucht kein init_run durchoeführt werden.						
MDS_control	MDS_c (siehe	ontrol schaltet die Anwesenheitskontrolle b Kapitel "Anwesenheitskontrolle und MDS -S	zw. die MDS -Ste Steuerung ").	euerung am A	SM ein oder aus		
	Wert	MDS-Steuerur	ng		ASM-Typ		
	0 Anwesenheitskontrolle ist ausgeschaltet. Die Variable ANZ_MSD_present zeigt keinen gültigen Wert an. alle 1 Anwesenheitskontrolle ist eingeschaltet. Die MDS-Steuerung ist abgeschaltet. Die Variable ANZ_MDS_present zeigt einen MDS im Übertragungsfenster eines SLG an. alle 2 Anwesenheitskontrolle ist eingeschaltet. Die MDS -Steuerung ist eingeschaltet und geschieht über die Anwesenheitskontrolle des MDS. Der NEXT-Befehl muss nach jeder MDS -Bearbeitung zum ASM geschickt werden. 454						
ECC_mode	Schaltet den ECC-Betrieb ein (True) oder aus (False). Beachten Sie, dass der ECC-Betrieb nur bei MOBY I zugelassen ist.						
RESET_long	Beim Befehl init_run werden alle INPUT-Parameter zum ASM übertragen. Dieses Bit muss beim Betrieb von MOBY U/D bzw. RF300 auf True gesetzt werden (MOBY_mode = 5).						

Notizen

Parameter "ASM_address" Wert = 256 (siehe Hardware-Konfiguration)
Parameter " ASM_channel " Wert = 1 für SLG1
Parameter " ASM_channel " Wert = 2 für SLG2
Parameter " command_DB_number " Wert = 47 für beide SLGs
Parameter " command_DB_address " Wert = 0 für SLG1
Parameter " command_DB_address " Wert = 50 für SLG2

i

Variable	Beschreibung							
MOBY_mode	Einstellung der MOBY-Betriebsart							
	Wert	Betriebsart	ASM-Typ					
	0	Default	-; reserviert für die Einstellung mit Schalter oder GSD-Parametrierung; verschiedene Anschaltungen ohne Schalter verstehen unter MOBY- mode = 0 die MOBY I-Betriebsart					
	1	MOBY I oder MOBY E (ohne MDS 507)	alle					
	4	MOBY I mit MDS 507	452; 454; 456					
	5	MOBY U/D bzw. RF300 - ohne Multitaghandling	475; 473; 452; 456; RF170C					
	6	res. für MOBY U - mit Multitaghandling (FB 55)	-					
	7	res. für MOBY D bzw. RF300 - mit Multitaghandling (FB 55)	-					
	8	MOBY I-Dialog	454; 452; 456					
	9	MOBY V	452					
	А	MOBY F mit MDS F1xx	475; 473: 452; 854; 850					
	В	MOBY F mit MDS F4xx	475; 473: 452; 854; 850					
	С	MOBY F (res. für MDS F2xx)	-					
	Zu bea Eine Ä	Zu beachten: Eine Änderung von MOBY_mode darf nur nach dem Einschalten eines ASM erfolgen.						
scanning_time	MOBY Scanni MDS-T nachfo Bit	I/V: ng_time ist die Abtastzeit für den MDS 507 von MOBY iypen kann hier der Wert 00 eingesetzt werden. Die Einstein Igende Grafik (siehe auch Projektierungshandbuch für State 7 6 5 4 2 1 0	und MOBY V. Bei allen anderen stellung der Abtastzeit (ABTA) zeigt SLG 44/MDS 507):					
	Beispie	11 = 10 s el: Abtastzeit von 1 Sekunde ergibt für den Parameter A	BTA = 81 hex.					

Notizen

i

MOBY U: Scanning_time beschreibt die Standby-Zeit für den MDS. Erhält der MDS vor Ablauf der scanning_time einen weiteren Befehl, so kann dieser sofort bearbeitet werden. Erhält der MDS einen Befehl nach Ablauf der scanning_time, so wird die Befehlsbearbeitung um die sleep_time des MDS verzögert. Eine scanning time sollte nur eingestellt werden, wenn
 der MDS mit mehreren Befehlen bearbeitet wird <i>und</i> der Bearbeitungsvorgang in einer minimalen Zeit abgeschlossen sein muss. 00 hex = keine Standby-Zeit (Default) 01 hex = 7 ms Standby-Zeit 02 hex = 14 ms Standby-Zeit : C8 hex = 1400 ms Standby-Zeit Beachte: Die scanning_time beeinflusst die Batterielebensdauer. Je größer die eingestellte scanning_time, desto kürzer die Batterielebensdauer. Genauere Berechnungen finden Sie im MOBY U-Handbuch für Projektierung, Montage und Service.
MOBY D bzw. RF300: 00 hex (reserviert) Dieses Byte ist bitweise kodiert. Es hat standardmäßig den Wert B#16#0. Es können damit besondere Steuerungen im ASM vorgenommen werden. Bit 7 6 5 4 3 2 1 0 Bit 7 6 5 4 3 2 1 0 Image: teaching of
MOBY U: Reichweitenbegrenzung
normale Sendeleistungreduzierte Sendeleistung05 hex = 0,5 m85 hex0A hex = 1,0 m8A hex0F hex = 1,5 m8F hex14 hex = 2,0 m91 hex19 hex = 2,5 m99 hex18 hex = 3,0 m9E hex23 hex = 3,5 mA3 hexMOBY D: HF-Leistung von 0,5 W bis 10 W in 0,25 W-Schritten02 hex = 0,5 W:10 hex = 4 W (default):28 hex = 10 W

Notizen

Parameter "**option_1**" Wert = 2 um das rote Blinken der ERR-LED am SLG mit einem

"Init_Run" zurückzusetzen ____

i

Variable	Beschreibung						
multitag	MOBY U/D bzw. RF300: Maximale Anzahl der parallel im Feld bearbeitbaren MDS. zugelassene Werte: 1						
field_ON_control	MOBY U: BERO-B "Antenne EIN/AUS	MOBY U: BERO-Betriebsart; automatisches Ein-/Ausschalten des Antennenfeldes. Der Befehl "Antenne EIN/AUS" wird von der BERO-Betriebsart überlagert.					
	00 hex	=	ohne BEROs; keine SLG-Synchronisation				
	01 hex	=	ein oder zwei BEROs Die BEROs sind logisch ODER verknüpft. Während der Zeit der Betätigung eines BEROs ist das Feld eingeschaltet.				
	02 hex	=	ein oder zwei BEROs. Der 1. BERO schaltet das Feld ein und der 2. BERO schaltet das Feld aus. Wenn zwei BEROs vorhanden sind <i>und</i> eine field_ON_time parametriert ist, wird das Feld automatisch ausgeschaltet, wenn der 2. BERO nicht innerhalb dieser BERO-Zeit schaltet. Ist keine field_ON_time parametriert, so bleibt das Feld bis zur Betätigung des 2. BERO eingeschaltet.				
	03 hex	=	SLG Synchronisation über Kabelverbindung aktivieren (siehe Handbuch fü Projektierung, Montage und Service für MOBY U)				
	MOBY D bzw. RF300: 00 hex (reserviert)						
field_ON_time	MOBY U: Zeit für E	BER	D-Betriebsart (field_ON_control = 02)				
	00 hex	=	Die Zeitüberwachung ist abgeschaltet. Für die Feldabschaltung wird der 2. BERO benötigt.				
	01 hex FF hex	=	1 255 s Einschaltzeit für das SLG-Feld				
	MOBY D: MDS-Typ						
	00 hex	=	I-Code 1 (z. B. MDS D139)				
	01 hex	= ISO-MDS					
	RF300: 00 hex (res	RF300: 00 hex (reserviert)					
reserved0	reserviert						

Notizen

Parameter "field_ON_time" Wert = 0 für MDS-Typ RF300 ______ Parameter "field_ON_time" Wert = 1 für MDS-Typ ISO______





Befehls- und Statuswort "BEST"

Im Befehls- und Statuswort sind die Steuerbits des FB 45 definiert. Das Befehls- und Statuswort mit den Variablen wird mit dem UDT 11 generiert. Die Variablen und die dazugehörigen relativen Adressen im UDT 11 sind im folgenden Bild wiedergegeben.

Steuerbits:

Mit diesen Boolschen Variablen startet der Anwender seine Befehle.





Steuerbits von Bit0 bis Bit7

Variable	Beschreibung
cancel	<i>True</i> = Unterbrechung eines laufenden Befehls bzw. einer Befehlskette. Die Variable ready
	wird daraufhin von dem FB 45 gesetzt.
	MOBY U/D bzw. RF300: Die Variable cancel ist nicht verfugbar. Ein Befehlsabbruch muss
command_start	<i>True</i> = Start eines Befehls bzw. einer Befehlskette.
repeat_command	<i>True</i> = Befehlswiederholung: Der zuletzt im ASMgespeicherte Befehl bzw. die Befehlskette wird erneut mit dem nächsten MDS abgearbeitet. Dabei wird die Befehlsbearbeitung zum MDS erst gestartet, nachdem der bereits bearbeitete MDS das Übertragungsfenster verlassen hat (ANZ_MDS_present = 0) und ein neuer MDS in das Übertragungsfenster des SLG eingetreten ist (ANZ_MDS_present: $0 \rightarrow 1$).
	<i>False</i> = Keine Befehlswiederholung bzw. die Befehlswiederholung wird angehalten, nachdem der mit dem Repeat-Befehl gestartete Befehl abgearbeitet ist. Bitte beachten Sie, dass dieses Bit vom Anwender rückgesetzt werden muss, um die Befehlswiederholung anzuhalten. Das Ergebnis der Befehlswiederholung wird durch Setzen von command_start durch den Anwender abgeholt.
	repeat_command wird nicht automatisch von dem FB 45 nach der Befehlsbearbeitung rückgesetzt. Die Befehle init run und cancel setzen die Variable repeat command zurück. Damit ist auch
	eine Befehlswiederholung im ASM unterbrochen. repeat_command kann mit dem nächsten command_start durch die Anwendung erneut gesetzt werden. Die Handhabung der Befehlswiederholung ist im Kapitel "Befehlswiederholung" beschrieben.
init_run	<i>True</i> = Neustart des Anschaltmoduls. Dabei wird auch der FB 45 rückgesetzt sowie das ASM neu parametriert. Alle Daten und Befehle im ASM gehen verloren. Dieses Bit muss im Neustart-OB (OB 100) für jeden MOBY-Kanal bzw. ASM gesetzt werden. Nach einem Ausfall des MOBY-ASM wird dem Anwender der Fehler error_MOBY=0F gemeldet. Der Anwender muss anschließend einen init_run ausführen.
	Beachte:
	 Beim Einspielen eines Parameter-Datenbausteins vom Programmiergerät zur SIMATIC ist das Bit init_run mit TRUE vorbelegt. Das hat eine automatische Durchführung eines Neustarts des ASM zur Folge.
	 Die Zeit f ür die Ausf ührung von init_run liegt normalerweise im Millisekundenbereich. Im Fehlerfall kann sich diese Zeit bis auf 15 s verl ängern.
ASM_failure	<i>True</i> = Das ASM ist ausgefallen. Dieses Bit wird vom Anwender im OB 122 gesetzt (siehe Kapitel "Baugruppenausfall programmieren"). Der FB 45 meldet daraufhin dem Anwender einen Fehler (error_FB = 09) und unterbricht einen laufenden Befehl. Wird der OB 122 vom Anwender nicht programmiert, so geht das AG bei einem ASM-Ausfall in den STOP-Zustand.
FB45_active	Der FB 45 bearbeitet gerade einen Befehl. Diese Variable wird beim Start des Befehls (command_start=True) gesetzt und bleibt solange aktiv, bis
	der FB 45 die letzte Quittung vom ASM erhalten hat
	das init_run-Bit gesetzt wurde
	das Cancel-Bit gesetzt wurde
	eine Fehlermeldung vom ASM gemeldet wurde
ANZ_next	Dieses Bit ist gesetzt, wenn der zuletzt ausgeführte Befehl ein NEXT-Befehl war.
ANZ reset	Dieses Bit zeigt an, dass der zuletzt ausgeführte Befehl ein RESET war. Der RESET-Befehl
	wurde vom Anwender mit "init run" gestartet.

Notizen

i

Steuerbits von Bit8 bis Bit15

Variable	Beschreibung
ANZ_MDS_present	Zeigt die Anwesenheit eines MDS im Übertragungsfenster des Read-Write-MDS an. ANZ_MDS_present wird nur angezeigt, wenn der INPUT-Parameter MDS_control (siehe Kapitel "Die INPUT-Parameter") vom Anwender gesetzt wurde. Beachten Sie, dass bei der Durchführung eines init_run die ANZ_MDS_present-Anzeige kurzzeitig verschwindet, auch wenn sich permanent ein MDS im Übertragungsfenster aufhält.
ANZ_cancel	Der zuletzt ausgeführte Befehl war eine Befehlsunterbrechung (cancel). Das Bit wird gesetzt, wenn das ASM über das zyklische Wort (siehe Kapitel "Zyklisches Steuerwort zwischen Master und MOBY-ASM") eine Cancel-Quittung anzeigt. Das Rücksetzen erfolgt automatisch durch den Start eines neuen Befehls.
ANZ_ECC	nur MOBY I: Bei eingeschaltetem ECC-Treiber (INPUT-Parameter "ECC-mode" = TRUE) zeigt das Bit an, dass die vom MDS gelesenen Daten korrigiert worden sind. ANZ_ECC ist <i>keine</i> Fehlermeldung, da die Daten in Ordnung sind. ANZ_ECC ist ein Hinweis, dass möglicherweise demnächst der gerade bearbeitete MDS-Speicher vollständig ausfällt.
reserved	z. Zt. nicht belegt
LR_bat	Dieses Bit hat nur Bedeutung, wenn bei MOBY I mit dem MDS 507 gearbeitet wird. Es zeigt eine leere Dialogbatterie des MDS 507 an. Bei allen anderen MDS kann dieses Bit jeden Zustand einnehmen.
battery_low	nur bei MOBY I/V mit RAM-MDS: Die Stützbatterie des RAM-MDS ist unterhalb des Schwellenwertes. Mit der verbleibenden Restkapazität kann bei Raumtemperatur zwar noch mehrere Monate gearbeitet werden, es wird jedoch empfohlen, die Batterie des MDS umgehend zu wechseln bzw. den MDS bei nicht wechselbarer Batterie auszutauschen.
error	Dieses Bit wird von dem FB 45 bei fehlerhaftem Abschluss eines Befehls gesetzt. Das error- Bit ist das Summenfehlerbit für alle auftretenden Fehler. Die genaue Fehlerursache steht dabei in den Variablen error_ MOBY, error_FB oder error_BUS (siehe Kapitel "Weitere Anzeigen" bzw. Kapitel "Fehlermeldungen und Fehlersuche"). Der erneute Start eines Befehls setzt das error-Bit wieder zurück.
ready	Fertigmeldung: Nachdem ready = TRUE gemeldet wurde, muss noch das error-Bit = FALSE abgefragt werden. Damit ist sichergestellt, dass der Befehl fehlerfrei bearbeitet wurde. Beachte: Für den Start von init_run oder cancel braucht das ready-Bit nicht gesetzt zu sein.

Notizen





Weitere Anzeigen

Die Anzeigen zeigen dem Anwender den Fortgang eines Befehls an. Fehleranalysen können einfach durchgeführt werden.

Variable	Beschreibung					
ASM_busy	Dieses Bit ist gesetzt, wenn das ASM einen Befehl bearbeitet. Normalerweise ist "ASM_busy" invertiert zu "ready". ASM_busy wird vom ASM über das zyklische Wort gemeldet (siehe im Kapitel "Zyklisches Steuerwort zwischen Master und MOBY-ASM" unter "ASM_busy_"). Wird mit dem automatischen Befehlsstart repeat_command gearbeitet, so zeigt dieses Bit die Bearbeitung eines neuen MDS mit dem auszuführenden Befehl an. MOBY U/D bzw. RF300: Keine Bedeutung; diese Variable ist immer false.					
command_rep_active	Das ASM führt gerade eine Befehlswiederholung aus. Das Bit wird als Antwort auf die Steuervariable repeat_command gesetzt. Nach einem init_run wird command_rep_active von dem FB 45 zunächst zurückgesetzt und erst verzögert wieder gesetzt, da hier der FB 45 zuerst die MOBY-Befehle zum ASM überträat.					
number_MDS	MOBY U/D bzw. RF300: Es wird die Anzahl der MDS angezeigt, die sich gegenwärtig im Übertragungsfenster befinden. Befinden sich mehr als 15 MDS im Feld, so bleibt die Anzeige number_MDS bei 0F hex stehen.					
error_MOBY	Dieser Fehler wurde vom ASM gemeldet. Die Anzeige dieses Fehlers erfolgt in der Regel auch auf der ERR-Leuchtdiode auf der ASM-Kanalanzeige (siehe Kapitel "Fehlermeldungen und Fehlersuche").					
error_FB	Fehlermeldung von dem FB 45 (siehe Kapitel "Fehlermeldungen und Fehlersuche")					
error_BUS	Die Übertragungsstrecke zwischen FB 45 und ASM meldet einen Fehler. In der Regel ist das ein PROFIBUS-Fehler. (siehe Kapitel "Fehlermeldungen und Fehlersuche"). Dieser Fehler wird von den System-Funktionen SFC 58/59 gemeldet.					
version_MOBY	Anzeige der Firmware-Version der MOBY-ASM. Der hier eingetragene Wert wird nach jedem Hochlauf der ASM aktualisiert. Er ist ASCII-kodiert. Beispiel: DBB 26 DBB 27 31 hex 30 hex → Version 1.0 "1" "0"					

Notizen



Alle weiteren Variablen des UDT 11 sind nur für den FB-internen Gebrauch. Sie dürfen in keinem Fall vom Anwender verändert werden

5.11 FB10 SLG_Steuerungsprogramm

SIEMENS



Im FB10 soll nun das Steuerungsprogramm zu den Steuerbefehlen eines SLGs und der Aufruf des FB45 programmiert werden. Zusätzlich soll noch die Anwesenheitszeit des Transponders erfasst werden.

Funktionsbaustein FB10 erstellen

Legen Sie einen neuen Funktionsbaustein FB10 an.

Eigenschaften - Funktionsbaustein 🛛 🛛 🔀						
Allgemein - Teil1 Allgemein - Teil 2 Aufrufe Attribute						
Name:	FB10 Multiinstanzfähig					
Symbolischer Name:	SLG_STEUERUNG					
Symbolkommentar:	Steuerbefehle zu einem SLG					
Erstellsprache:	FUP					
Projektpfad:	RFID_RF180C_IEC\SIMATIC 300(1)\CPU 315F-2 PN/DP\S7- Programm(2)\Bausteine\FB10					
Speicherort des Projekts:	C:\Program Files\Siemens\Step7\s7proj\RFID_R_6					
Erstellt am:	Code Schnittstelle 08.12.2009 18:57:22					
Zuletzt geändert am:	08.12.2009 20:07:04 08.12.2009 19:04:41					
Kommentar:	SLG-Baustein					
OK	Abbrechen Hilfe					

IN Variablen anlegen

Inhalt von: 'Ungebung\Schnittstelle\IN'							
i⊐-∎ IN	~		Name	Datentyp	Adresse	Anfangswert	
🔤 slg_start		12	slg_start	Bool	0.0	FALSE	
🔤 mds_present		12	mds_present	Bool	0.1	FALSE	
──``⊡ slg_error		12	slg_error	Bool	0.2	FALSE	
🔤 slg_ready	≡	12	slg_ready	Bool	0.3	FALSE	
Isig_option_1		12	slg_option_l	Byte	1.0	B#16#0	
🔤 slg_reset		12	slg_reset	Bool	2.0	FALSE	
──`⊡ slg_iso		12	slg_iso	Bool	2.1	FALSE	
- 🖾 Params_DB		12	Params_DB	Int	4.0	0	
🖾 Params_ADDR	\mathbf{v}	12	Params_ADDR	Int	6.0	0	

OUT Variablen anlegen

Inhalt von: 'Umgebung\Schnittstelle\OUT'								
≓⊶ ⊒ - о∪т	^	Name	Datentyp	Adresse	Anfangswert			
🖾 command_start		🔟 command_start	Bool	8.0	FALSE			
─`⊡ option_1		🔚 option_l	Byte	9.0	B#16#0			
🖾 field_on_time		🔚 field_on_time	Byte	10.0	B#16#0			
🖾 init_run		🛅 init_run	Bool	11.0	FALSE			
🖾 mds_time	=	🖲 mds_time	Time	12.0	T#OMS			





STAT Variablen anlegen

	Inhalt von: 'Ungebung\Schnittstelle\STAT'							
	^		Name	Datentyp	Adresse	Anfangswert		
- 🖻 fml	_	12	fml	Bool	16.0	FALSE		
19 fm2		12	fm2	Bool	16.1	FALSE		
` E fm3		12	fm3	Bool	16.2	FALSE		
` ` I fm4		12	fm4	Bool	16.3	FALSE		
19 fm5		12	fm5	Bool	16.4	FALSE		
13 fn6		12	fm6	Bool	16.5	FALSE		
🖾 mds_zeit		12	mds_zeit	Time	18.0	T#OMS		

TEMP Variablen anlegen

Inhalt von: 'Umgebung\Schnittstelle\TE								
🕂 🖅 STAT	~		Name		Datentyp	j	Adresse	
Ė~ 4∭- TEMP	_	12	initl		Bool		0.0	
🖓 🖾 initl		12	init2		Bool	1	0.1	
🖓 🖾 init2		12	init3		Bool		0.2	
🖾 🖬 init3	~	ъ						

Netzwerke 1 bis 3

FB10 : SLG-Baustein

Netzwerk 1 : Kommando Start



Netzwerk 2 : SLG rücksetzen









Netzwerke 4 bis 8

Netzwerk 4 : RF300-Modus



Netzwerk 5 : init_run



Netzwerk 6 : Anwesenheitszeit MDS an SLG



Netzwerk 7: SLG1 MDS-Zeitauswertung



Netzwerk 8 : FB45 Aufruf



Hinweis zu den Netzwerken 7 und 8:

Die Einschaltverzögerung **TON (SFB4)** im Netzwerk 7 und der **FB45** Aufruf im Netzwerk 8 werden als Multiinstanz eingefügt. Nach dem Einfügen ins Programm den Baustein mit rechter Maustaste anklicken und "Ändern in Multiinstanz-Aufruf" wählen. Danach im Fenster den Namen der Multiinstanz (Siehe NW7 bzw. NW8 ohne #) eingeben.

FB10 speichern und schließen

5.12 FB1 Steuerungsprogramm



Mit den fertigen Bausteinen kann jetzt das Steuerungsprogramm erstellt werden.

Aufgabenstellung:

In unserem Beispielprogramm sollen zwei Befehlsketten mit je zwei Aufträgen abgearbeitet werden.

Mit den **START_SLG1 (E0.0)** wird die Befehlskette des ersten SLG ausgeführt. Erst soll die MDS-Status-Information gelesen werden und danach soll am SLG1 ein Schreib-Befehl durchgeführt werden. Wir schreiben mit den SLG1 8 Byte die im DB48 ab DBB0 abgelegt aufs MDS. Mit den **RESET_SLG1** (E0.1) wird im Fehlerfall (LED am SLG1 blinkt rot) der Fehler zurückgesetzt und ein "Init run Befehl" zum Rücksetzen des ersten SLG ausgeführt.

Mit den **RF300_ISO** (E02) soll am ersten SLG (RF310R) ein Umschalten zwischen MDS-Typen z.B. RF360T oder ISO-Transpnder z.B. Moby D ISO durchgeführt werden.

Mit den **START_SLG2** (E1.0) wird die Befehlkette des zweiten SLG ausgeführt. Erst soll die MDS-Diagnose-Informationen gelesen werden und danach soll am SLG2 ein Lese-Befehl durchgeführt werden. Wir lesen mit dem SLG2 8 Byte aus dem Transponder und schreiben diese in den DB48 ab DBB50.

Mit den **RESET_SLG2** (E1.1) wird im Fehlerfall (LED am SLG2 blinkt rot) der Fehler zurückgesetzt und ein "Init_run Befehl" zum Rücksetzen des zweiten SLG ausgeführt.

Zusätzlich soll noch an den SLGs die Anwesenheitszeit der Transponder erfasst werden.

Symboltabelle ergänzen

Symbol	Adresse 🛆	Datentyp	Kommentar
START_SLG1	E 0.0	BOOL	Befehlsstart des SLG1
RESET_SLG1	E 0.1	BOOL	SLG1 Fehler rücksetzen
RF300_ISO	E 0.2	BOOL	Wert 0 = RF300, Wert 1 = ISO
START_SLG2	E 1.0	BOOL	Befehlsstart des SLG2
RESET_SLG2	E 1.1	BOOL	SLG2 Fehler rücksetzen
AWZ_SLG1	MD 40	TIME	Anwesenheitszeit des Transpnders am SLG1
AWZ_SLG2	MD 50	TIME	Anwesenheitszeit des Transpnders am SLG2
STATUS_SLG_1	VAT 1		Variablentabelle des SLG1
STATUS_SLG_2	VAT 2		Variablentabelle des SLG2



Funktionsbaustein FB1 erstellen

Legen Sie einen neuen Funktionsbaustein FB1 an.

Eigenschaften - Funkti	onsbaustein		X					
Allgemein - Teil1 Allgemei	Allgemein - Teil1 Allgemein - Teil 2 Aufrufe Attribute							
Name:	FB1	31 Multiinstanzfähig						
Symbolischer Name:	TEUERUNGSPROGRAMM							
Symbolkommentar:	Steuerungsprogramm mit E	}austeinen						
Erstellsprache:	FUP							
Projektpfad:	rojektpfad: RFID_RF180C_IEC\SIMATIC 300(1)\CPU 315F-2 PN/DP\S7- Programm(2)\Bausteine\FB1							
Speicherort des Projekts:	C:\Program Files\Siemens\Step7\s7proj\RFID_R_6							
	Code	Schnittstelle						
Erstellt am:	08.12.2009 19:05:17							
Zuletzt geändert am:	08.12.2009 20:24:43	08.12.2009 19:29:03						
Kommentar:	Steuerungsprogramm	~	J					
ОК		Abbrechen Hilfe						



TEMP Variablen anlegen

	In	Inhalt von: 'Umgebung\Schnittstelle\TEMP'						
IN IN	~	Name	Datentyp	Adresse				
		🕲 init_runl	Bool	0.0				
IN_OUT		🔟 init_run2	Bool	0.1				
🛨 🖅 STAT	≡	12						
+ - TEMP								

Netzwerk 1 bis 2

Der FB10 wird als Multiinstanz-Baustein eingefügt.



TIA Ausbildungsunterlage

Ausgabestand: 01/2010

#init runl -



Netzwerk 3

Netzwerk 3 : Auftrag oder Befehl zu SLG1 aufrufen



 Bitmuster fu	ır	INIT-Betchl;	Modus fu	r END	SET-	-ANT,	MDS-STATUS	STC-	STA
 Anzahl der s	11	schreibenden/	lesenden	Date	n in	Byt.e	e e		

- Anzahl der zu schreibenden/lesenden Daten in Bytes
 Anfangsadresse auf IDD3; Bndadresse bei INTI;KW/Jahr für MDS-STATUS
 Mummar des DAT DB: Daten DB für MDS Daten
 Zeiger auf das Anfangsvort im DAT DB
 MDS Befehl:l=schreiben,2=lesen,3=init,4=slg-status,8=end,A=set-ant,B=mds-status
 Bitawster für INIT-Befehl; Modus für KND,SKT-ANT,MDS-STATUS,SLG-STATUS
 Anzahl der zu schreibenden/lesenden Daten in Bytes
 Anfangsadresse auf MDS; Endadresse bei INIT;KW/Jahr für MDS-STATUS
 Nummer des DAT DB; Daten DB für MDS Daten
 Zeiger auf das Anfangsvort im DAT DB



Netzwerk 4 bis 5

Netzwerk 4 : Steuerbefehle zu SLG2 aufrufen



Symbolinformation:

Symbolinformation:

-
-
-
-
-
-
-
-
-

-- Befehlsstart des SLG2 -- Anwesenheit eines MDS

- -- Fehler während der Befehlsbearbeitung -- Der Befehl bzw. Befehlskette ist ausgeführt
- -- SLG2 Fehler rücksetzen
 - -- Set: Startsignal für Befehl bzw. Befehlskette
 - -- Input: RESET-Befehl Option 1 -- Input: BERO-Zeit
- -- Anwesenheitszeit des Transpnders am SLG2

Netzwerk 5 : Set: SLG2 ASM rücksetzen und neu parametrieren



"SLG_PARAMETER".SLG_2.init_run

DB45.DBX69.3

-- Set: ASM rücksetzen und neu parametrieren



Netzwerk 6

Netzwerk 6 : Auftrag oder Befehl zu SLG2 aufrufen





FB1 speichern und schließen

5.13 OB1 Programmaufruf



Doppelklicken Sie den OB1 im Projektfenster oder öffnen Sie die Objekteigenschaften des OB1 und geben Sie den Symbolischen Namen und Symbolkommentar ein.

Eigenschaften - Organisationsbaustein					
Allgemein - Teil1 Allgemein - Teil2 Aufrufe Attribute					
Name:	OB1				
Symbolischer Name:	PROGRAMMAUFRUF				
Symbolkommentar:	Steuerungsprogramm aufrufen				
Erstellsprache:	FUP				
Projektpfad:	RFID_RF180C_IEC\SIMATIC: Programm(2)\Bausteine\OB1	300(1)\CPU 315F-2 PN/DP\S7-			
Speicherort des Projekts:	C:\Program Files\Siemens\Step7\s7proj\RFID_R_6				
Erstellt am:	Code 05.12.2009 16:34:07	Schnittstelle			
Zuletzt geändert am:	08.12.2009 19:29:24	15.02.1996 16:51:12			
Kommentar:	Steuerungsprogramm aufrufen				
ОК		Abbrechen Hilfe			

OB1 öffnen und FB1 mit DB1 aufrufen

OB1 : Steuerungsprogramm aufrufen	
Kommentar:	
Netzwerk 1: FBl aufrufen	
Konnentar:	
DB1 "STEUERUNGSPROGRAMM " EN ENO	
Symbolinformation: STEUERUNGSPROGRAMM FB1	Steuerungsprogramm mit Bausteinen

Fenster mit der Abfrage zum Generieren des DB1 durch Klick auf "Ja" bestätigen.

OB1 speichern und schließen

Nun kann das Programm in die Steuerung geladen und getestet werden.

5.14 Variablentabelle STATUS_SLG_1



Operand	Symbol	Anzeigeformat Statuswert S		Steuerwert	
// Cancel					
DB45.DBX 19.0	"SLG_PARAMETER".SLG_1.cancel	BOOL	false		
// Command Start					
DB45.DBX 19.1	"SLG_PARAMETER".SLG_1.command_start	BOOL	false		
// System Start Up)				
DB45.DBX 19.3	"SLG_PARAMETER".SLG_1.init_run	BOOL	false		
// Ready					
DB45.DBX 18.7	"SLG_PARAMETER".SLG_1.ready	BOOL	l true		
// Presence of a N	4DS				
DB45.DBX 18.0	"SLG_PARAMETER".SLG_1.ANZ_MDS_present	BOOL	false		
MD 40	"AWZ_SLG1"	ZEIT	T#5s21ms		
// Error					
DB45.DBX 18.6	"SLG_PARAMETER".SLG_1.error	BOOL	false		
// Errors					
DB45.DBB 22	"SLG_PARAMETER".SLG_1.error_MOBY	HEX	B#16#00		
DB45.DBB 23	"SLG_PARAMETER".SLG_1.error_FB	HEX	B#16#00		
// MOBY Comman	d				
DB47.DBB 0	"AUFTRAG".Kanal_1_Befehl[1].command	HEX	B#16#4B		
DB47.DBB 1	"AUFTRAG".Kanal_1_Befehl[1].sub_command	HEX	B#16#01		
DB47.DBW 2	"AUFTRAG".Kanal_1_Befehl[1].length	DEZ	18		
DB47.DBW 4	"AUFTRAG".Kanal_1_Befehl[1].address_MDS	HEX	VV#16#0000		
DB47.DBW 6	"AUFTRAG".Kanal_1_Befehl[1].DAT_DB_number	DEZ	49		
DB47.DBW 8	"AUFTRAG".Kanal_1_Befehl[1].DAT_DB_address	DEZ	0		
DB48.DBB 0	"S_L_DATEN".Daten[1]	ZEICHEN	'R'	'R'	
DB48.DBB 1	"S_L_DATEN".Daten[2]	ZEICHEN	'F'	'F'	
DB48.DBB 2	"S_L_DATEN".Daten[3]	ZEICHEN	Ψ	Ψ.	
DB48.DBB 3	"S_L_DATEN".Daten[4]	ZEICHEN	'D'	'D'	
DB48.DBB 4	"S_L_DATEN".Daten[5]	ZEICHEN	'T'	'T'	
DB48.DBB 5	"S_L_DATEN".Daten[6]	ZEICHEN	'E'	'E'	
DB48.DBB 6	"S_L_DATEN".Daten[7]	ZEICHEN	'S'	'S'	
DB48.DBB 7	"S_L_DATEN".Daten[8]	ZEICHEN	'T'	'T'	
DB48.DBB 50	"S_L_DATEN".Daten[51]	ZEICHEN	B#16#00		
DB48.DBB 51	"S_L_DATEN".Daten[52]	ZEICHEN	B#16#00		
DB48.DBB 52	"S_L_DATEN".Daten[53]	ZEICHEN	B#16#00		
DB48.DBB 53	"S_L_DATEN".Daten[54]	ZEICHEN	B#16#00		
DB48.DBB 54	"S_L_DATEN".Daten[55]	ZEICHEN	B#16#00		
DB48.DBB 55	"S_L_DATEN".Daten[56]	ZEICHEN	B#16#00		
DB48.DBB 56	"S_L_DATEN".Daten[57]	ZEICHEN	B#16#00		
DB48.DBB 57	"S_L_DATEN".Daten[58]	ZEICHEN	B#16#00		
DB45.DBB 16	"SLG_PARAMETER".SLG_1.field_ON_time	HEX	B#16#00		
DB47.DBB 10	"AUFTRAG".Kanal_1_Befehl[2].command	HEX	B#16#01		
DB47.DBB 11	"AUFTRAG".Kanal_1_Befehl[2].sub_command	HEX	B#16#00		
DB47.DBW 12	"AUFTRAG".Kanal_1_Befehl[2].length	DEZ	8		
DB47.DBW 14	"AUFTRAG".Kanal_1_Befehl[2].address_MDS	HEX	VV#16#0000		
DB47.DBW 16	"AUFTRAG".Kanal_1_Befehl[2].DAT_DB_number	DEZ	48		
DB47.DBW 18	"AUFTRAG".Kanal_1_Befehl[2].DAT_DB_address	DEZ	0		

5.15 Variablentabelle STATUS_SLG_2



Operand	Symbol	Anzeigeformat	Statuswert	Steuerwert
// Cancel	·			
DB45.DBX 69.0	"SLG_PARAMETER".SLG_2.cancel	BOOL	false	
// Command Start				
DB45.DBX 69.1	"SLG_PARAMETER".SLG_2.command_start	BOOL	false	
// System Start U	p			
DB45.DBX 69.3	"SLG_PARAMETER".SLG_2.init_run	BOOL	false	
// Ready				
DB45.DBX 68.7	"SLG_PARAMETER".SLG_2.ready	BOOL	📘 true	
// Presence of a l	MDS			
DB45.DBX 68.0	"SLG_PARAMETER".SLG_2.ANZ_MDS_present	BOOL	false	
MD 50	"AWZ_SLG2"	ZEIT	T#3s994ms	
// Error				
DB45.DBX 68.6	"SLG_PARAMETER".SLG_2.error	BOOL	false	
// Errors				
DB45.DBB 72	"SLG_PARAMETER".SLG_2.error_MOBY	HEX	B#16#00	
DB45.DBB 73	"SLG_PARAMETER".SLG_2.error_FB	HEX	B#16#00	
// MOBY Comman	nd		-	
DB47.DBB 50	"AUFTRAG".Kanal_2_Befehl[1].command	HEX	B#16#4B	
DB47.DBB 51	"AUFTRAG".Kanal_2_Befehl[1].sub_command	HEX	B#16#02	
DB47.DBW 52	"AUFTRAG".Kanal_2_Befehl[1].length	DEZ	18	
DB47.DBW 54	"AUFTRAG".Kanal_2_Befehl[1].address_MDS	HEX	VV#16#0000	
DB47.DBW 56	"AUFTRAG".Kanal_2_Befehl[1].DAT_DB_number	DEZ	49	
DB47.DBW 58	"AUFTRAG".Kanal_2_Befehl[1].DAT_DB_address	DEZ	18	
DB48.DBB 0	"S_L_DATEN".Daten[1]	ZEICHEN	'R'	
DB48.DBB 1	"S_L_DATEN".Daten[2]	ZEICHEN	'F'	
DB48.DBB 2	"S_L_DATEN".Daten[3]	ZEICHEN	Т	
DB48.DBB 3	"S_L_DATEN".Daten[4]	ZEICHEN	'D'	
DB48.DBB 4	"S_L_DATEN".Daten[5]	ZEICHEN	'T'	
DB48.DBB 5	"S_L_DATEN".Daten[6]	ZEICHEN	'E'	
DB48.DBB 6	"S_L_DATEN".Daten[7]	ZEICHEN	'S'	
DB48.DBB 7	"S_L_DATEN".Daten[8]	ZEICHEN	'T'	
DB48.DBB 50	"S_L_DATEN".Daten[51]	ZEICHEN	'R'	
DB48.DBB 51	"S_L_DATEN".Daten[52]	ZEICHEN	'F'	
DB48.DBB 52	"S_L_DATEN".Daten[53]	ZEICHEN	Ψ	
DB48.DBB 53	"S_L_DATEN".Daten[54]	ZEICHEN	'D'	
DB48.DBB 54	"S_L_DATEN".Daten[55]	ZEICHEN	'T' 	
DB48.DBB 55	"S_L_DATEN".Daten[56]	ZEICHEN	'E'	
DB48.DBB 56	"S_L_DATEN".Daten[57]	ZEICHEN	'S'	
DB48.DBB 57	"S_L_DATEN".Daten[58]	ZEICHEN	'T'	
DB47.DBB 60	"AUFTRAG".Kanal_2_Befehl[2].command	HEX	B#16#02	
DB47.DBB 61	"AUFTRAG".Kanal_2_Befehl[2].sub_command	HEX	B#16#00	
DB47.DBW 62	"AUFTRAG".Kanal_2_Betehl[2].length	DEZ	8	
DB47.DBW 64	"AUFTRAG".Kanal_2_Betehl[2].address_MDS	HEX	VV#16#0000	
D847.D8W 66	"AUFTRAG".Kanal_2_Betehl[2].DAT_DB_number	DEZ	48	
DB47.DBW 68	"AUFTRAG".Kanal_2_Betehl[2].DAT_DB_address	DEZ	50	

5.16 Symboltabelle

Q	

Symbol Adresse A		Datentyp		Kommentar		
SLG_PARAMETER	DB	45	DB	45	hier findet man die Parameter zu den SLGs	
AUFTRAG	DB	47	DB	47	hier werden die Kommandos eingegeben	
S_L_DATEN	DB	48	DB	48	hier werden die Daten des Schreib-Lese Auftrags eingetragen	
MDS_STATUS	DB	49	DB	49	hier werden die MDS-Status Informationen abgelegt	
SLG_STATUS	DB	50	DB	50	hier werden die SLG-Status Informationen abgelegt	
START_SLG1	E	0.0	BOO	Ĺ	Befehlsstart des SLG1	
RESET_SLG1	E	0.1	BOO	L	SLG1 Fehler rücksetzen	
RF300_ISO	E	0.2	BOO	Ĺ	Wert 0 = RF300, Wert 1 = ISO	
START_SLG2	E	1.0	BOO	L	Befehlsstart des SLG2	
RESET_SLG2	E	1.1	BOO	L	SLG2 Fehler rücksetzen	
STEUERUNGSPROGR	FB	1	FB	1	Steuerungsprogramm mit Bausteinen	
SLG_STEUERUNG	FB	10	FB	10	Steuerbefehle zu einem SLG	
MOBY FB	FB	45	FB	45		
AUFTRAG_BEFEHL	FC	11	FC	11	Baustein für einen Auftrag bzw. Befehl	
AWZ_SLG1	MD	40	TIME		Anwesenheitszeit des Transpnders am SLG1	
AWZ_SLG2	MD	50	TIME		Anwesenheitszeit des Transpnders am SLG2	
PROGRAMMAUFRUF	OB	1	OB	1	Steuerungsprogramm aufrufen	
NEUSTART	OB	100	OB	100	Programmierung von Neustart und Wiederanlauf	
TON	SFB	4	SFB	4	Generate an On Delay	
RDREC	SFB	52	SFB	52	Read a Process Data Record	
WRREC	SFB	53	SFB	53	Write a Process Data Record	
MOBY Param_d	UDT	11	UDT	11	Parametrier DB	
MOBY CMD_d	UDT	21	UDT	21	Command DB Normaladressierung	
MOBY SLG-Status_d	UDT	111	UDT	111	MOBY: SLG Status Daten	
MOBY P MDS-Status	UDT	261	UDT	261	MOBY P: MDS Status Daten	
MOBY P MDS-Status	UDT	271	UDT	271	MOBY P: MDS Status Daten	
MOBY P SLG-Status_d	UDT	281	UDT	281	MOBY P: SLG Status Daten	
STATUS_SLG_1	VAT	1			Variablentabelle des SLG1	
STATUS_SLG_2	VAT	2			Variablentabelle des SLG2	

5.17 Bausteinordner

Objektname	Symbolischer Name	Erstellsprache	Größe im Arbeitsspei…	Тур	Versio
Systemdaten				SDB	
➡ OB1	PROGRAMMAUFRUF	FUP	72	Organisationsbaustein	0.1
➡ OB100	NEUSTART	AWL	52	Organisationsbaustein	0.1
➡ FB1	STEUERUNGSPROGRA	FUP	1242	Funktionsbaustein	0.1
➡ FB10	SLG_STEUERUNG	FUP	346	Funktionsbaustein	0.1
📮 FB45	MOBY FB	AWL	7846	Funktionsbaustein	1.3
■ FC11	AUFTRAG_BEFEHL	FUP	98	Funktion	0.1
➡ DB1		DB	972	Instanzdatenbaustei	0.0
🕩 DB45	SLG_PARAMETER	DB	136	Datenbaustein	0.1
🕩 DB47	AUFTRAG	DB	136	Datenbaustein	0.1
🕩 DB48	S_L_DATEN	DB	1060	Datenbaustein	0.1
🕩 DB49	MDS_STATUS	DB	72	Datenbaustein	0.1
➡ DB50	SLG_STATUS	DB	92	Datenbaustein	0.1
➡ UDT11	MOBY Param_d	AWL		Datentyp	1.3
➡ UDT21	MOBY CMD_d	AWL		Datentyp	1.3
➡ UDT111	MOBY SLG-Status_d	AWL		Datentyp	1.3
UDT261	MOBY P MDS-Status 0_d	AWL		Datentyp	1.3
UDT271	MOBY P MDS-Status 1_d	AWL		Datentyp	1.3
UDT281	MOBY P SLG-Status_d	AWL		Datentyp	1.3
TATUS_SLG_1	STATUS_SLG_1			Variablentabelle	1.3
STATUS_SLG_2	STATUS_SLG_2			Variablentabelle	1.3
📮 SFB4	TON	AWL		Systemfunktionsbau	1.0
📮 SFB52	RDREC	AWL		Systemfunktionsbau	1.0
05050	WDDEC	43.0		Contraction lation also a	1.0

5.18 Datensicht des DB49

🖙 @DB49 "MDS_STATUS" RFID_RF180C_IEC\SIMATIC 300(1)\CPU 315F-2 PN/DP\\DB49_ONLINE					
Adresse	Name	Тур	Anfangswert	Aktualwert	Kommentar
0.0	MDS_Status0.reserved0	BYTE	B#16#0	B#16#01	
1.0	MDS_Status0.status_info	BYTE	B#16#0	B#16#00	Modus MDS-Status
2.0	MDS_Status0.UID[1].Byte_1_4	DWORD	DW#16#0	DW#16#00000000	MDS-Nummer (unique identifier)
6.0	MDS_Status0.UID[1].Byte_5_8	DWORD	DW#16#0	DW#16#557AFEFA	
10.0	MDS_Status0.MDS_type	BYTE	B#16#0	B#16#02	MDS-Typ
11.0	MDS_Status0.Lock_state	BYTE	B#16#0	B#16#00	Schreibschutzstatus EEPROM
12.0	MDS_Status0.reserved1[1]	BYTE	B#16#0	B#16#00	
13.0	MDS_Status0.reserved1[2]	BYTE	B#16#0	B#16#00	
14.0	MDS_Status0.reserved1[3]	BYTE	B#16#0	B#16#00	
15.0	MDS_Status0.reserved1[4]	BYTE	B#16#0	B#16#00	
16.0	MDS_Status0.reserved1[5]	BYTE	B#16#0	B#16#00	
17.0	MDS_Status0.reserved1[6]	BYTE	B#16#0	B#16#00	
18.0	MDS_Statusl.reserved0	BYTE	B#16#0	B#16#02	
19.0	MDS_Statusl.status_info	BYTE	B#16#0	B#16#00	Modus MDS-Status
20.0	MDS_Status1.UID[1].Byte_1_4	DWORD	DW#16#0	DW#16#00000000	MDS-Nummer (unique identifier)
24.0	MDS_Statusl.UID[1].Byte_5_8	DWORD	DW#16#0	DW#16#557BA57F	
28.0	MDS_Status1.LFD	BYTE	B#16#0	B#16#0D	Leistungsflussdichte: Beziehung zwischen Grenzwert um
29.0	MDS_Status1.FZP	BYTE	B#16#0	B#16#00	Fehlerzähler passiv (Ruhefehlerzähler)
30.0	MDS_Status1.FZA	BYTE	B#16#0	B#16#03	Fehlerzähler aktiv (Fehler während Kommunikation)
31.0	MDS_Status1.ANWZ	BYTE	B#16#0	B#16#FF	Anwesenheitszähler
32.0	MDS_Statusl.reserved1[1]	BYTE	B#16#0	B#16#00	
33.0	MDS_Statusl.reserved1[2]	BYTE	B#16#0	B#16#00	
34.0	MDS_Statusl.reserved1[3]	BYTE	B#16#0	B#16#00	